IBM

@server

iSeries

Printer Device Programming

Version 5

SC41-5713-05

IBM

@server

iSeries

Printer Device Programming

Version 5

SC41-5713-05

Sixth Edition (September 2002)

This edition replaces SC41-5713-04. This edition applies only to reduced instruction set computer (RISC) systems.

Contents

About Printer Device Programming (SC41-5713)

This book provides information about printing elements of OS/400, printer file and spooling support for printing operation, Advanced Function Printing™ (AFP™), as well as printing in a network environment.

Who should read this book

This book is intended for the application programmer and system programmer. Some system operators may also find this book helpful in understanding printer fields and spooling support. To use this book, you should be familiar with the different types of printers your business uses, as well as job and queue management.

Conventions and terminology used in this book

The commands, parameters, and displays shown in this book reflect the intermediate assistance level (*INTERMED). The level of assistance (ASTLVL) is specified in the user profile. The following is a list of values for the assistance levels:

- *SYSVAL
- *BASIC
- *INTERMED
- *ADVANCED

Prerequisite and related information

Use the iSeries Information Center as your starting point for looking up iSeries technical information.

You can access the Information Center two ways:

- From the following Web site:

  http://www.ibm.com/eserver/iseries/infocenter
- From CD-ROMs that ship with your Operating System/400 order:

  iSeries Information Center, SK3T-4091-02. This package also includes the PDF versions of iSeries manuals, *iSeries Information Center: Supplemental Manuals*, SK3T-4092-01, which replaces the Softcopy Library CD-ROM.

The iSeries Information Center contains advisors and important topics such as Java™, TCP/IP, Web serving, secured networks, logical partitions, clustering, CL commands, and system application programming interfaces (APIs). It also includes links to related IBM® Redbooks™ and Internet links to other IBM Web sites such as the Technical Studio and the IBM home page.

With every new hardware order, you receive the *iSeries Setup and Operations CD-ROM*, SK3T-4098-01. This CD-ROM contains IBM @server iSeries Access for Windows and the EZ-Setup wizard. iSeries Access offers a powerful set of client and server capabilities for connecting PCs to iSeries™ servers. The EZ-Setup wizard automates many of the iSeries setup tasks.

For other related information, see the "Bibliography" on page 543.

iSeries Navigator

IBM iSeries Navigator is a powerful graphical interface for managing your iSeries servers. iSeries Navigator functionality includes system navigation, configuration, planning capabilities, and online help to guide you through your tasks. iSeries Navigator makes operation and administration of the server easier and more productive and is the only user interface to the new, advanced features of the OS/400 operating system. It also includes Management Central for managing multiple servers from a central server.

You can find more information on iSeries Navigator in the iSeries Information Center and at the following Web site:

http://www.ibm.com/eserver/iseries/navigator/

Using Print Services Facility™ for OS/400® (PSF/400)

Beginning with OS/400 V3R1, the advanced function printing (AFP) function is a separately orderable feature of OS/400 called Print Services Facility for OS/400 (PSF/400).

The OS/400 printing function continues to support line printers and a subset of IBM IPDS™ printers and print functions when PSF/400 is not installed.

Full support for all IPDS printers is provided by the integrated AFP printing function. The printing function used to process application output is determined by the device description of the target printer. Only printers defined as DEVTYPE(*IPDS) and AFP(*YES) (both specified in the printer device description) are controlled by the AFP printing function.

When is PSF/400 Required?

The PSF/400 feature is required when:
- Any of the following IBM printers are used:
  - IBM 3820 Page Printer
  - IBM 3825 Page Printer
  - IBM 3827 Page Printer
  - IBM 3828 Advanced Function MICR Printer
  - IBM 3829 Advanced Function Printer :
  - IBM 3831 Page Printer (in 3835 emulation mode; available only in Japan)
  - IBM 3835-001 Page Printer
  - IBM 3835-002 Advanced Function Printer
  - IBM 3900 Advanced Function Printer
  - IBM 3130 Advanced Function Page Printer
  - IBM 3160 Advanced Function Page Printer
  - IBM 3935 Advanced Function Page Printer
  - IBM InfoPrint 60
  - IBM InfoPrint 62
  - IBM InfoPrint 70
  - IBM InfoPrint 2000
  - IBM InfoPrint 3300
  - IBM InfoPrint 4000

- Any printer that has the DEVTYPE parameter specified as *IPDS and the AFP parameter specified as *YES.
- Distributed print to IPDS or Hewlett-Packard** PCL4 or PCL5 printers attached to Print Services Facility for OS/2® (PSF for OS/2) LAN print server.
- The Advanced Function Printing Utilities/400 licensed program is to be used.

When is PSF/400 Optional?

When the following IPDS printers or IPDS printer models are installed, PSF/400 is optional. These printers can be driven by the original OS/400 printing subsystem. However, PSF/400 is required if the functions provided by the AFP subsystem are to be used.

- IBM 3112 and 3116 Page Printers
- IBM 3812 and 3816 Page Printers
- IBM 3912 and 3916 Page Printers
- IBM 3930 Page Printer
- IBM LaserPrinter 4028
- IBM 4224 and 4234 Printers
- IBM 4230 Print
- IBM 4247 Printer
- IBM Network Printers (4312, 4317, and 4324)
- IBM 6408 Printer 1
- IBM 6412 Printer
- IBM InfoPrint 20
- IBM InfoPrint 32
- IBM InfoPrint 21
- IBM InfoPrint 40
- IBM 4400
- IBM 6400

Printer File Parameter Keywords Requiring PSF/400

The following shows printer file parameters and DDS keywords that affect print formatting and handling. When used to print on IPDS printers, they require PSF/400.

Printer File Parameters

- AFP characters (AFPCHARS)
- Back margin (BACKMGN)
- Back overlay (BACKOVL
- Coded font (CDEFNT)
- Corner staple (CORNERSTPL)
- Device type (*AFPDS, *LINE, *AFPDSLINE)
- Edge stitch (EDGESTITCH)
- Font character set (FNTCHRSET)
- Font resolution (FNTRSL)
- Front margin (FRONTMGN)
- Form definition (FORMDF)

- Front overlay (FRONTOVL)
- IPDS pass through (IPDSPASTHR(*YES|*NO))
- Multi-up (MULTIUP REDUCE(*NONE))
- Output bin (OUTBIN)
- Page definition (PAGDFN)
- Saddle stitch (SADLSTITCH)
- Table reference characters (TBLREFCHR)
- User-defined data (USRDFNDTA (IPDSPASTHR(*YES|*NO)))
- User-defined data (USRDFNDTA(USRRCSLIBL (lib1 lib2 lib3 lib4))
- User resource library list (USRRSCLLIBL)

DDS Keywords

- Box (BOX)
- Coded font (CDEFNT)
- Data Stream Command (DTASTMCMD)
- Document Index Tag (DOCIDXTAG)
- End Page (ENDPAGE)
- End Page Group (ENDPAGGRP)
- Font character set (FNTCHRSET)
- Graphic data file (GDF)
- Invoke medium map (INVMMAP)
- Line (LINE)
- Overlay (OVERLAY)
- Page segment (PAGSEG)
- Position (POSITION)
- Start Page Group (STRPAGGRP)
- Text rotate (TXTRTT)
- Z fold (ZFOLD)
- Force (FORCE)
- Duplex (DUPLEX)
- Outbin (OUTBIN)

PrintManager/400

PrintManager/400 remains part of the OS/400 operating system. Print Services Facility for OS/400 is required if you use PrintManager/400 to place data on AS/400® spool or select page and form definition resources (AFP print objects as used on System/390® and RS/6000®).

How to send your comments

Your feedback is important in helping to provide the most accurate and high-quality information. If you have any comments about this book or any other iSeries documentation, fill out the readers' comment form at the back of this book.

- If you prefer to send comments by mail, use the readers' comment form with the address that is printed on the back. If you are mailing a readers' comment form from a country other than the United States, you can give the form to the local IBM branch office or IBM representative for postage-paid mailing.

- If you prefer to send comments by FAX, use either of the following numbers:
  - United States, Canada, and Puerto Rico: 1-800-937-3430
  - Other countries: 1-507-253-5192
- If you prefer to send comments electronically, use one of these e-mail addresses:
  - Comments on books:

    RCHCLERK@us.ibm.com
  - Comments on the iSeries Information Center:

    RCHINFOC@us.ibm.com

Be sure to include the following:
- The name of the book or iSeries Information Center topic.
- The publication number of a book.
- The page number or topic of a book to which your comment applies.

Summary of Changes

A vertical line (|) to the left of the text indicates a change or | addition.

Part 1. Introduction to printing on the iSeries server

Chapter 1. Understanding printing elements of the iSeries server

How is printing done on this system?

Where did my printed output go?

How can I control where my printouts go?

How can I print something on a different printer?

To know the answer to these and other printing questions you need to:
- Understand the elements that make printing happen.
- Understand which elements have precedence over other elements. For example; your job description has precedence over your user profile in determining which printer your output will print on.

After reading through this chapter and the examples in Appendix A, "Examples of Working with Printing Elements", you will be familiar with the elements that make printing happen and you will be able to:
- Display your user profile.
- Change your user profile.
- Create an output queue.
- Move spooled files from one output queue to a different output queue.
- Start printers printing.
- Stop printers from printing.
- Assign a printer to print spooled files from an output queue that it is currently not assigned to.

Note to Readers:

There are examples in Appendix A, "Examples of Working with Printing Elements" on page 349 of this guide that show you how the printing elements work together to enable you to manage your printing work.

Learning about printing elements and terminology

Many elements within the iSeries server contribute to creating and handling data that you want printed. To understand those elements, you should become familiar with the terminology that describes them.

Read through the following list to familiarize yourself with the elements, but do not worry about trying to memorize each definition. A thorough explanation of the elements and how they interact with each other will follow.

Printing Elements
  Element Definition

Spooled File
  A file that holds output data waiting to be printed.

A spooled file is usually the result of a system program, an application program being run, or the Print key being pressed.

Output Queue
An object that contains a list of spooled files to be printed.

Output queues can receive spooled files from more than one application program and more than one user.

Printer Writer
A function of the operating system that writes (sends) the spooled file from an output queue to a printer.

In most cases the application program sends the spooled file to an output queue first. Then the printer writer program sends it to a printer.

Remote Writer
A function of the operating system that writes (sends) the spooled file from an output queue to another system. The other system is usually connected through a communications line.

Print Devices
The physical printers that can be attached to the iSeries server.

Print devices (printers) should not be confused with the printer writer program or printer files.

Printer Files
Files that describe how the system is to operate on data as it passes between a program and a printer.

A printer file has many parameters. The spooling parameter (SPOOL) determines if your output goes to an output queue or directly to a printer. The device (DEV) parameter is the name of the printer your output is printed on. The output queue (OUTQ) parameter is the name of the output queue your spooled files are sent to.

Job Description
A system object, made up of many parameters, that defines how a job is to be processed. Once a job begins, the parameters in the job description become the attributes of the job. For more detailed information about job descriptions and job attributes, see the Work Management topic in the iSeries Information Center.

Printer device (PRTDEV) and output queue (OUTQ) are the two parameters that help determine where your output will go.

Workstation Description
Information collected from the device description for the display. Two of the device description parameters, printer device (PRTDEV) and output queue (OUTQ), help determine where your output will go. Go to "Understanding the workstation description" on page 18 for more information about the workstation description.

User Profile
An object with a unique name that contains the user's password, the list of special authorities assigned to a user, and the objects the user owns.

A user profile has many parameters. Printer device (PRTDEV) and output queue (OUTQ) are the two parameters that help determine where your output will go.

System Values
Values that control information for operating certain parts of the system. System administrators can change system values to redefine the working environment.

The system value most important to printing is the default system printer.

Messages
Responses from the system to a user of that system.

When printing operations are started, the system often asks the user to respond to a forms alignment message. Also, an application program could send a message asking the user to perform a variety of checks before printing starts. It is important to remember to respond to messages when requesting printing. Failure to respond to messages can prevent a printer from printing.

Learning how printing is done on the iSeries server

The diagram below illustrates many of the elements that control how printing work originates and flows through the system to a printer. Study the diagram to become familiar with the elements. Information on the following pages provides more detail and explains the relationships between the elements.

AS/400

| Application Program Runs or a Print Key Is Pressed |

An application program is run or the Print key is pressed.

Printer File

System-Supplied or User-Created

Application program looks at printer file to determine output queue name if spooling is to be used, or printer device name if spooling is not used.

Spooled File Created

Spooled files are the result of programs being run or Print key being used.

Output Queue

Spooled File
Spooled File
Spooled File

Output queue holds spooled files. Output queues are created by the system or by the user using the Create Output Queue (CRTOUTQ) command.

Printer Writer Program

A system-supplied program takes spooled files from output queues and sends them to a printer.

Printer

A physical printer is configured to the system by Autoconfiguration or manually using the CRTDEVPRT command.

RV2H327-0

As the diagram illustrates, there are many elements (such as printer files and output queues) and many actions (creating spooled files, starting printer writer programs) involved in processing your printing requests.

The remainder of the chapter explains the elements and the processes that combine to produce printed output.

Learning where printed output goes

The next three topics in this chapter discuss **printer files, spooled files**, and **output queues**.

Generally, when the user initiates a print request or runs a program, a printer file is accessed, a spooled file is generated, and that spooled file is sent to an output queue. The next three topics discuss the relationship between printer files, spooled files, and output queues.

Learning about printer files

Note: The information in this chapter on printer files is quite brief and intended only to introduce the printer file concept to you. For more detailed information on printer file support, go to Chapter 2, "Printer file support" on page 41.

Printers attached to the iSeries server are supported by the operating system through **printer files**. Printer files describe how the system is to operate on the data as it passes between your application program and a printer.

A printer file handles every request for printing. You can create your own printer files by using the Create Printer File (CRTPRTF) command, or you can use system-provided printer files.

Note: There is one exception to this. It applies only to the Print key when the display station has an attached remote workstation controller. In this situation, the printer that is named in the device description for that display station receives the printed version of the screen. The remote work station controller, not the iSeries server, handles this.

Printer files contain many parameters that tell the system how the output should be formatted, what font to use for the printed output, whether to print on both sides of the page, and more. The parameters that control how your output is handled and where it goes are:

1 Spool the data (SPOOL)

2 Device (DEV)

3 Spooled output queue (OUTQ)

4 Spooled File Owner(SPLFOWN)

If another printer file is not specified, the printer uses the default printer file. The default printer file for the system is QSYSPRT. The following page identifies the SPOOL, DEV, OUTQ, and SPLFOWN parameters on the default printer file, QSYSPRT.

```
                     Display Spooled File

File  . . . . . :   QPDSPFD                    Page/Line   1/1
  Control . . . . .                              Columns    1 - 78
  Find  . . . . . .

  2/09/98                Display File Description

 DSPFD Command Input
    File  . . . . . . . . . . . . . . . . . : FILE      QSYSPRT
      Library . . . . . . . . . . . . . . . :           *LIBL
    Type of information . . . . . . . . . . : TYPE      *ALL
    File attributes . . . . . . . . . . . . : FILEATR   *ALL
    System  . . . . . . . . . . . . . . . . : SYSTEM    *LCL
  File Description Header
    File  . . . . . . . . . . . . . . . . . : FILE      QSYSPRT
    Library . . . . . . . . . . . . . . . . :           QSYS
    Type of file  . . . . . . . . . . . . . :           Device
    Device type . . . . . . . . . . . . . . :           Printer
    Auxiliary storage pool ID . . . . . . . :           01
  Device File Attributes
    Externally described file . . . . . . . :           No
    File level identifier . . . . . . . . . :           0980109013920
```

```
        Creation date  . . . . . . . . . . . . . :          01/09/98
        Text 'description' . . . . . . . . . . . : TEXT      System non-describ
      *************************************************************************
❶   Spool the data . . . . . . . . . . . . . : SPOOL      *YES
      *************************************************************************
Maximum devices . . . . . . . . . . . . . . :          1
    User specified DBCS data . . . . . . . . : IGCDTA    *NO
    Maximum file wait time . . . . . . . . . : WAITFILE  *IMMED
    Share open data path . . . . . . . . . . : SHARE     *NO
    Record format level check . . . . . . . . : LVLCHK   *NO
    Number of record formats . . . . . . . . :          1
    User buffer length . . . . . . . . . . . :          0
    Number of devices . . . . . . . . . . . . :          1
    Separate indicator area . . . . . . . . . : INDARA   No
    Coded character set identifier . . . . . . : CCSID        0
 Printer Attributes
      *************************************************************************
❷   Device . . . . . . . . . . . . . . . . . : DEV       *JOB
      *************************************************************************
    Printer device type . . . . . . . . . . . : DEVTYPE   *SCS
     Page size                                  PAGESIZE
     Length  . . . . . . . . . . . . . . . . :              34
     Width . . . . . . . . . . . . . . . . . :             132
     Measurement Method  . . . . . . . . . . :          *ROWCOL
    Lines per inch . . . . . . . . . . . . . : LPI       6
    Characters per inch . . . . . . . . . . . : CPI       10
    Front margin  . . . . . . . . . . . . . . : FRONTMGN  *DEVD
    Back margin . . . . . . . . . . . . . . . : BACKMGN   *FRONTMGN
    Overflow line number  . . . . . . . . . . : OVRFLW    30
    Fold records  . . . . . . . . . . . . . . : FOLD      *NO
    Degree of page rotation . . . . . . . . . : PAGRTT    *AUTO
    Hardware justification  . . . . . . . . . : JUSTIFY   0
    Print on both sides . . . . . . . . . . . : DUPLEX    *NO
    Defer Write . . . . . . . . . . . . . . . : DFRWRT    *YES
    Unprintable character action               RPLUNPRT
     Replace character . . . . . . . . . . . :          *YES
       Replacement character . . . . . . . . . :          ' '   X'40'
    Print text  . . . . . . . . . . . . . . . : PRTTXT    *JOB
    Align page  . . . . . . . . . . . . . . . : ALIGN     *NO
    Control character . . . . . . . . . . . . : CTLCHAR   *NONE
    Channel values  . . . . . . . . . . . . . : CHLVAL    *NORMAL
    Fidelity . . . . . . . . . . . . . . . . : FIDELITY  *CONTENT
    Printer quality . . . . . . . . . . . . . : PRTQLTY   *STD
    Form feed . . . . . . . . . . . . . . . . : FORMFEED  *DEVD
    Source drawer . . . . . . . . . . . . . . : DRAWER    1
    Output bin  . . . . . . . . . . . . . . . : OUTBIN    *DEVD
    Font                                        FONT
     Identifier  . . . . . . . . . . . . . . :          *CPI
     Point size  . . . . . . . . . . . . . . :          *NONE
    Character identifier  . . . . . . . . . . : CHRID     *CHRIDCTL
    Decimal format  . . . . . . . . . . . . . : DECFMT    *JOB
    Font character set  . . . . . . . . . . . : FNTCHRSET *FONT
    Coded font . . . . . . . . . . . . . . . . : CDEFNT   *FNTCHRSET
    Table Reference Characters  . . . . . . . : TBLREFCHR *NO
    AFP Chars . . . . . . . . . . . . . . . . : AFPCHARS  *NONE
    Page definition . . . . . . . . . . . . . : PAGDFN    *NONE
    Form definition . . . . . . . . . . . . . : FORMDF    *NONE
    Form type . . . . . . . . . . . . . . . . : FORMTYPE  *STD
    Pages per side  . . . . . . . . . . . . . : MULTIUP   1
    Reduce output . . . . . . . . . . . . . . : REDUCE    *TEXT
    Unit of measure . . . . . . . . . . . . . : UOM       *INCH
    Front side overlay  . . . . . . . . . . . : FRONTOVL  *NONE
    Back side overlay . . . . . . . . . . . . : BACKOVL   *FRONTOVL
    IPDS pass through . . . . . . . . . . . . : IPDSPASTHR *DEVD
    Convert line data . . . . . . . . . . . . : CVTLINDTA *NO
    User resource library list  . . . . . . . : USRRSCLIBL *DEVD
    Corner staple . . . . . . . . . . . . . . : CORNERSTPL *NONE
```

```
   Edge stitch                                      EDGESTITCH
    Reference edge . . . . . . . . . . . . . . :               *NONE
   Font resolution . . . . . . . . . . . . . : FNTRSL     *DEVD
   Saddle stitch                              : SADLSTITCH
    Reference edge   . . . . . . . . . . . . :               *NONE
   DBCS extension characters . . . . . . . . : IGCEXNCHR  *YES
   DBCS character rotation . . . . . . . . . : IGCCHRRTT  *NO
   DBCS characters per inch  . . . . . . . . : IGCCPI     *CPI
   DBCS SO/SI spacing  . . . . . . . . . . . : IGCSOSI    *YES
   DBCS Coded font . . . . . . . . . . . . . : IGCCDEFNT  *SYSVAL
Spooling Description
 *******************************************************************************
 3    Spooled output queue  . . .    : OUTQ       *JOB
 *******************************************************************************
   Max spooled output records  . . . . . . . : MAXRCDS        100000
   Spooled output schedule . . . . . . . . . : SCHEDULE   *FILEEND
   Copies  . . . . . . . . . . . . . . . . . : COPIES       1
   Page range to print                         PAGERANGE
     Starting page . . . . . . . . . . . . . :                      1
     Ending page . . . . . . . . . . . . . . :               *END
   File separators . . . . . . . . . . . . . : FILESEP      0
   Hold spooled file . . . . . . . . . . . . : HOLD       *NO
   Save spooled file . . . . . . . . . . . . : SAVE       *NO
   Output priority (on OUTQ) . . . . . . . . : OUTPTY     *JOB
   User data . . . . . . . . . . . . . . . . : USRDTA     *SOURCE
 *******************************************************************************
 4 Spooled file owner  . . . . . . . . . . . : SPLFOWN    *CURUSRPRF
 *******************************************************************************
   User defined option . . . . . . . . . . . : USRDFNOPT *NONE
   User defined data . . . . . . . . . . . . : USRDFNDTA *NONE
   User defined object                         USRDFNOBJ
     Object  . . . . . . . . . . . . . . . . :               *NONE
       Library . . . . . . . . . . . . . . . :
     Object type . . . . . . . . . . . . . . :
Record Format List
                   Record  Format Level
   Format      Fields  Length  Identifier
   QSYSPRT        0       0  0000000000000
     Text . . . . . . . . . . . . . . . . . :
   Total number of formats  . . . . . . . . . :          1
   Total number of fields . . . . . . . . . . :          0
   Total record length  . . . . . . . . . . . :          0
```

SPOOL parameter = *YES

*YES is the default value for the SPOOL parameter.

When the SPOOL parameter is set to *YES, the output from an application
program (a spooled file) is sent to an output queue (OUTQ). When SPOOL = *YES,
the system looks at the OUTQ parameter in the printer file to find out which
output queue (OUTQ) to send the spooled file to. For example, the OUTQ value in
your printer file could be OUTQ1.

However, in the default printer file, QSYSPRT, the value specified is *JOB. This
means that the QSYSPRT printer file tells the system to look at your **job's OUTQ
attribute** to determine the name of the output queue (OUTQ).

SPOOL parameter = *NO

When the SPOOL parameter is set to *NO, the output from an application program
is sent directly to a printer. When SPOOL = *NO, the system looks at the DEV
parameter in the printer file to find out which printer to send the output to. For
example, the DEV value in your printer file could be PRT01.

However, in the default printer file, QSYSPRT, the value specified is *JOB. This means that the QSYSPRT printer file tells the system to look at your **PRTDEV attribute** to determine the name of the printer device (DEV).

Device (DEV) parameter

The device parameter indicates the name of a printer device description. If SPOOL = *NO is specified, the device parameter identifies the printer device used to produce the printed output. If SPOOL = *YES is specified, the device (DEV) parameter is ignored unless *DEVD is specified for the output queue parameter. In that case, the default output queue for the specified printer is used for the spooled files. For more information about output queues, see "Learning about output queues (CRTOUTQ)" on page 12.

Spooled file owner (SPLFOWN) parameter

The SPLFOWN parameter specifies which user profile owns a file. The spooled file can be owned by the current job, a QPRTJOB for the current user, a group profile for the current user, or the group profile of the current job's user.

For example, for debugging purposes, you might want the current job to spool and own all service dumps, instead of an individual user. To accomplish this, you could set the parameter to the special value of *JOB for the QPSRVDMP printer file.

Output queue (OUTQ) parameter

The output queue parameter indicates which output queue your spooled files are sent to. If you have a program that creates large print jobs you might consider sending them to an output queue that will hold those spooled files until most of your printer work for the day is done. Doing this can help users who have lots of small jobs get their jobs printed in a reasonable amount of time. For more information about output queues, see "Learning about output queues (CRTOUTQ)" on page 12.

Learning about spooling and spooled files

> **Note:** The information in this chapter on spooling is quite brief and intended only to introduce the spooling concept to you. A more detailed explanation of spool support is provided in Chapter 3, "Spool support".

Spooling is a system function that saves data in a database file for later processing or printing. This data, which is saved and eventually printed, is called a *spooled file*.

When spooling is used, spooled files) are created from the application program, a system program, or the pressing of the Print key. These files are put on disk in places called *output queues*.

The diagram below shows an iSeries server with three application programs. When these application programs are run, they take information from a *printer file*.

Information in the printer file determines if the output should be spooled, and which output queue the spooled file is sent to.

A key concept to remember is that you can create and use multiple printer files and multiple output queues to accomplish your printing needs. Also, two different application programs can send spooled files to the same output queue. This is shown in the bottom portion of the diagram.

```
┌─────────────────────────────────────────────────────────────────────┐
│ AS/400                                                               │
│                                                                     │
│                                          ┌──────────────────────┐   │
│                                          │ Output Queue         │   │
│   ┌──────────────────┐                   │ Name = OUTQONE       │   │
│   │ Application Program │                 ├──────────────────────┤   │
│   │ Using a Printer File │                │                      │   │
│   │ Called PRTFONE    │◄──┐              │ Spooled File         │   │
│   └──────────────────┘   │ ┌──────────────┤ Spooled File         │   │
│            ▲             └─┤ Printer File │ Spooled File         │   │
│            │               ├──────────────┤                      │   │
│            │               │ Name = PRTFONE │                    │   │
│            │               │ OUTQ = OUTQONE │                    │   │
│            │               └──────────────┘                      │   │
│                                                                     │
│   ┌──────────────────┐   ┌──────────────┐                          │
│   │ Application Program │◄──┤ Printer File │                         │
│   │ Using a Printer File │  ├──────────────┤                         │
│   │ Called PRTFTWO    │   │ Name = PRTFTWO │                        │
│   └──────────────────┘   │ OUTQ = OUTQTWO │                        │
│                          └──────────────┘                          │
│                                          ┌──────────────────────┐   │
│                                          │ Output Queue         │   │
│                                          │ Name = OUTQTWO       │   │
│   ┌──────────────────┐                   ├──────────────────────┤   │
│   │ Application Program │                 │                      │   │
│   │ Using a Printer File │                │ Spooled File         │   │
│   │ Called PRTFTHREE  │   ┌──────────────┤ Spooled File         │   │
│   └──────────────────┘◄──┤ Printer File │ Spooled File         │   │
│                          ├──────────────┤                      │   │
│                          │ Name = PRTFTHREE                     │   │
│                          │ OUTQ = OUTQTWO                       │   │
│                          └──────────────┘      RV2H326-0         │
└─────────────────────────────────────────────────────────────────────┘
```

Spooling and application programs

Almost all application programs that generate printed output make use of the
spooling support provided with the iSeries server. Whether spooling support is
requested is determined by specifying SPOOL = *YES or SPOOL = *NO on the
SPOOL parameter of a printer file.

By creating and using your own printer file instead of using a system-supplied
printer file, you can specify the printing control instructions that your application
program receives.

You can use the Create Printer File (CRTPRTF) command to create your own
printer file.

For more detailed information and examples about printer files, see Chapter 2,
"Printer file support".

Spooling and the print key

Using the Print key to capture an image of a screen almost always results in a
spooled file being created (SPOOL = *YES must be specified in the printer file
named in the workstation device description). Unless the value has been changed,
the default value for the SPOOL attribute in the QSYSPRT printer file is *YES.
When the Print key is pressed, the system looks at the OUTQ parameter in the
QSYSPRT printer file to determine which output queue to send the spooled file to.

Why use spooling to manage your printing workload?

Spooling (SPOOL = *YES) has several advantages over direct output (SPOOL =
*NO in the printer file):

- The user's display station remains available for work.

- Other users can request printing work without having to wait for the printer to become available.
- If special forms are required, you can have the spooled files sent to a special output queue and printed at a time when the printer is not busy.
- Since disk operations are much faster than printers, the system is used efficiently.

Learning about output queues (CRTOUTQ)

Output queues are objects, defined to the system, that provide a place for spooled files to wait until they are printed.

Output queues are created in two ways:
- By the user
- By the system

User-created output queues
You can create an output queue using the Create Output Queue (CRTOUTQ) command. On the prompt display, specify the name for the output queue to create. The output queue will be in the library identified by the library prompt. You can create as many output queues as you want.

System-created output queues
When a printer is configured to the system, either manually or through automatic configuration, the system creates an output queue for that printer.

System-created output queues are commonly called device output queues and have the same name as the printer device.

For example: when you configure a printer using the Create Device Description (Printer) (CRTDEVPRT) command, if you assign the printer name PRT01 in the DEVD parameter, the system creates an output queue named PRT01.

How to determine your output queue
If none of the IBM-supplied default values for the system have been changed, you can identify your output queue by displaying the system value QPRTDEV. Your output queue has the same name as the value shown for the system printer.

Why use multiple output queues?
Spooled files are created when application programs are run. If you do not want the spooled files to print right away, you can have them sent to an output queue that currently does not have a printer assigned to it.

For example: let us assume that you have only one printer available. One of your application programs creates a job that has 600 pages of printed output. Since all users are using the same printer, you do not want to print the 600-page job until everyone has finished working for the day.

One solution is to create two separate output queues. One output queue receives the spooled files from the application program that creates the 600 pages of printed output. The other output queue receives the spooled files from the jobs run by other users, as shown in the diagram below.
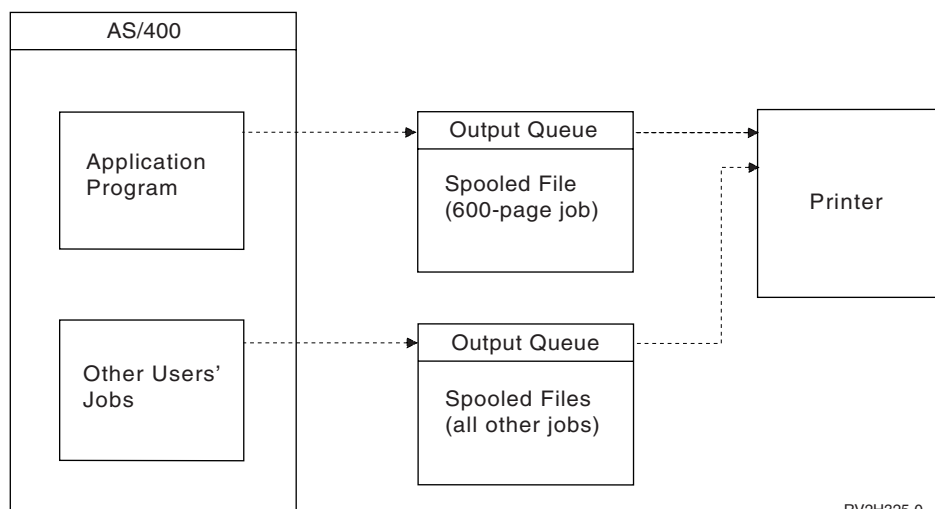
```
 ┌──────────────────────────────────────┐
 │ AS/400                               │
 │                                      │
 │  ┌─────────────┐      ┌─────────────────┐      ┌─────────────┐
 │  │             │      │ Output Queue    │      │             │
 │  │ Application │------→│─────────────────│------→│             │
 │  │ Program     │      │ Spooled File    │      │  Printer    │
 │  │             │      │ (600-page job)  │      │             │
 │  └─────────────┘      └─────────────────┘      └─────────────┘
 │                                      
 │  ┌─────────────┐      ┌─────────────────┐
 │  │             │      │ Output Queue    │
 │  │ Other Users'│------→│─────────────────│
 │  │ Jobs        │      │ Spooled Files   │
 │  │             │      │ (all other jobs)│
 │  └─────────────┘      └─────────────────┘
 └──────────────────────────────────────┘
                                            RV2H325-0
```

The program that creates the 600-page job sends the spooled file to a specific
output queue. That output queue does not have a printer assigned to it. Therefore,
the 600-page spooled file has to wait until a printer is assigned; meanwhile, the
spooled files that are in the other output queue can be printed.

Multiple output queues can also be used with deferred printing. To print a large
spooled file which exceeds the current limit for the printer's output queue, the
printer can be assigned to an output queue without any limit. Another solution is
to set the maximum spooled file size to print during a specified time. For example,
a maximum spooled file size of 100 pages could be set from 08:00:00 to 17:30:00
hours. During this time, only spooled files of 100 or less pages would print. After
5:30 PM, any spooled file prints. Spooled files which are too large are placed in
deferred status (*DFR) until they can be printed. See "Controlling printing from an
output queue by spooled file size" on page 131 and "Controlling multiple output
queues" on page 136 for more information on using multiple output queues with
deferred printing.

Controlling print activity

In the first part of this chapter you read about **spooled files** and **output queues**.
These elements are identified by **1** in the diagram below.

The remainder of this chapter discusses the elements that control or direct the
printing activity.

- The printer writer program.

  This element is identified by **2** in the diagram below.

  Note: The Start Remote Writer (STRRMTWTR) command is similar to the Start
  Printer Writer (STRPRTWTR) command. The STRRMTWTR command
  provides function that allows spooled files to be sent to a remote system
  for printing. For more information, see Chapter 4, "Remote System
  Printing" on page 159.

- Printer files, job attributes, user profiles, workstation descriptions, job
  descriptions, and system values.

  These elements are identified by **3** in the diagram below.

```
  ┌─┐
  │1│
  └─┘
      ┌────────────────────┐
      │  Output Queue      │
      ├────────────────────┤
      │                    │
      │  Spooled  File     │
      │  Spooled  File     │
      │  Spooled  File     │
      │                    │
      └────────────────────┘

  ┌─┐                          ┌─┐
  │2│                          │3│
  └─┘                          └─┘
  ┌──────────────────┐   ┌──────────────────────────┐
  │                  │   │  Printer  File           │
  │                  │   │  Job  Attributes         │
  │ Printer  Writer  │   │  Job  Description        │
  │ Program          │   │  User  Profile           │
  │ (system-supplied)│   │  Workstation  Description│
  │                  │   │  System  Value           │
  └──────────────────┘   └──────────────────────────┘

              ┌──────────────────┐
              │                  │
              │     Printer      │
              │                  │
              └──────────────────┘
                                        RV2H322-1
```
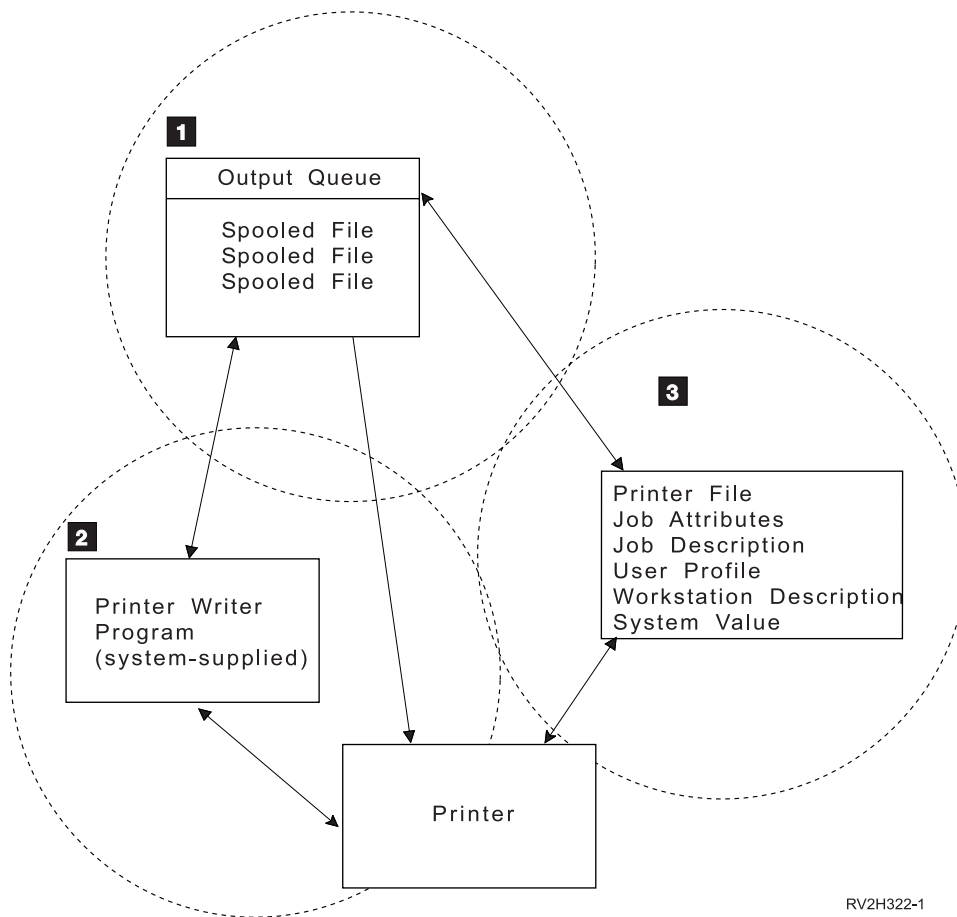
This diagram is designed to show you, by use of the joined circles, that the elements that control printing are interrelated; to produce printed output, the attributes that are common to all the elements must be correctly matched.

These important attributes, common to the elements that control printing, are:

- Output queues
- Printer devices

Learning about the printer writer program

The printer writer program is a system-supplied program that enables the user to work with the printers attached to the iSeries server.

Note: Do not confuse the printer writer program with an actual printer device or a printer file. The printer device is a physical printer and the printer writer is the program that allows you to assign an actual printer device to an output queue and select spooled files from the output queue to be printed.

The diagram below illustrates how the **printer writer program** interacts with the output queue and the physical printer to take spooled files from an output queue and send them to a printer.

AS/400

Application
Program
Runs or a
Print Key
Is Pressed

Printer File

System-Supplied
or User-Created

Spooled File
Created

Output Queue

Spooled File
Spooled File
Spooled File

Printer Writer
Program
(system-
supplied)

Physical
Printer

Device
Description

Printed
Output

RV2H324-0
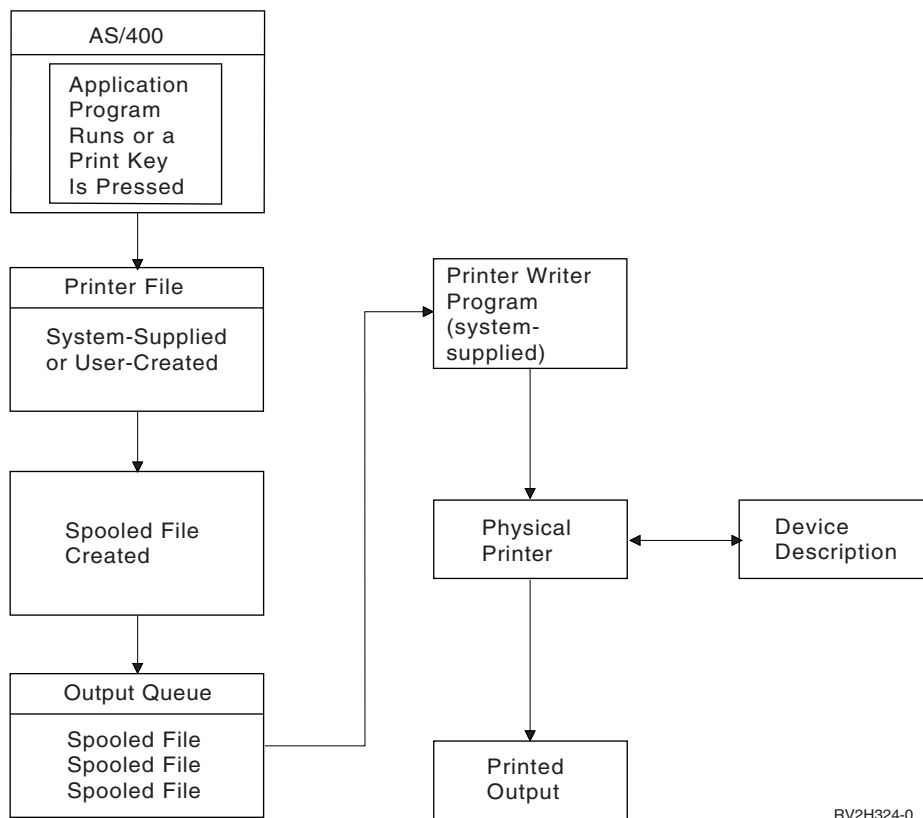
Printer device descriptions have to be created for each printer attached to the
system. If you use automatic configuration, this is done for you by the system,
with the exception of printers attached to an ASCII work station controller. Or you
can use the Create Device Description (Printer) (CRTDEVPRT) command to assign
a name to each printer.

The Start Printer Writer (STRPRTWTR) command and the Work with Writers
(WRKWTR) command provide you with the ability to assign any configured
printer to any output queue.

Notes:

1. Even though the name of the command (Work with Writers) indicates you are
   working with printer writers, you are actually using the printer writer program
   to make a match between an output queue and a physical printer.

2. If your printer stops for any reason (out of paper, for example), the iSeries
   server does not automatically assign a different printer to continue printing the
   jobs in the output queue that your printer was assigned to. You have to
   manually assign another printer to that output queue.

Understanding the hierarchy of the printing elements

The elements that control printing have a defined hierarchy. The following diagram
shows that hierarchy. The system looks first for the output queue and print device
in the printer file. The hierachy is broken in to two stages: job initiation and job
run time.

Assuming that the method used to start the job does not override the output
queue or printer device values specified in the device description, the order
followed for job initiation is: job description, user profile, workstation description,

and system value. The results of the job initiation stage are called **job attribues**.These job attributes are characteristics that define how the system should process a job.

For job run time, the order followed is printer file and job attributes. In the following diagram you can see how these stages work. For additional information about job attributes and manipulating job attributes, see the Work Management topic in the online Information Center.

```
┌──────────────────────────┐        ┌──────────────────────────┐
│     Job Description       │        │     Job Description       │
├──────────────────────────┤        ├──────────────────────────┤
│  Ouput queue *USRPRF      │        │   Printer device          │
│                          │        │      *USRPRF              │
└──────────────────────────┘        └──────────────────────────┘
            │                                    │
            ▼                                    ▼
┌──────────────────────────┐        ┌──────────────────────────┐
│      User Profile         │        │      User profile         │
├──────────────────────────┤        ├──────────────────────────┤
│   Output queue            │        │   Printer device          │
│     *WRKSTN               │        │     *WRKSTN               │
└──────────────────────────┘        └──────────────────────────┘
            │                                    │
            ▼                                    ▼
┌──────────────────────────┐        ┌──────────────────────────┐
│ Workstation description   │        │ Workstation description   │
├──────────────────────────┤        ├──────────────────────────┤
│   Output queue            │        │   Printer device          │
│     *DEV                  │        │     *SYSVAL               │
└──────────────────────────┘        └──────────────────────────┘
            │                                    │
            │                                    ▼
            │                        ┌──────────────────────────┐
            │                        │      System value         │
            │                        ├──────────────────────────┤
            │                        │   Printer device          │
            │                        │     PRT01                 │
            │                        └──────────────────────────┘
            │                                    │
┌──────────────────────────┐        ┌──────────────────────────┐
│      Printer file         │        │     Job attributes        │
├──────────────────────────┤        ├──────────────────────────┤
│ Spool the data *YES or *NO│        │ Spool output queue (*DEV) │
│ Spool output queue (*JOB) │        │  Printer device (PRT01)   │
│ Printer device (*JOB)     │        │                          │
└──────────────────────────┘        └──────────────────────────┘
                                                 │
                                                 ▼
                                     ┌──────────────────────────┐
                                     │ If the values supplied    │
                                     │ for these elements by     │
                                     │ IBM have not been         │
                                     │ changed, your spooled     │
                                     │ output files go to the    │
                                     │ output queue named        │
                                     │ PRT01.                    │
                                     └──────────────────────────┘
```

RBAFT502-1

Note: If the printer identified in the system value does not exist on your iSeries server, the output goes to output queue QPRINT.

The OUTQ and printer selection proccess occurs in two stages:
- Stage 1: Job initiation
- Stage 2: Job run time

Stage 1: Job initiation

During stage 1, the following order is used to determine the output queue (OUTQ) and printer device (PRTDEV) values: job description, user profile, workstation description, and system values. The system stores the values for OUTQ and PRTDEV locally, as job attributes, with the job to be used during the run-time stage.

The IBM-supplied default job description, containing default job attributes, is QDFTJOBD.

For more detailed information about job attributes, see the Work Management topic in the iSeries Information Center.

Understanding the user profile

On the following page is an example of a **user profile** with the attribute values set to the system-supplied defaults.

To become a user of the system you must have a user profile. In most cases, someone having security officer authority adds new users to the system. A **user profile** is created for each new user added to the system. The user profile has three parameters (highlighted on the following page) that provide information to determine where the user's printed output will go.

1   Job Description

2   Output Queue

3   Printer Device

Job description: In the example user profile, this parameter and its value of QDFTJOBD (item **1** in the following example) mean that when the user requests a printing job, the system scans the QDFTJOBD job description to determine which output queue (OUTQ) and printer device (PRTDEV or DEV) it should use. If the job description has the value of *USRPRF specified for these parameters, the system comes back to the user profile and continues to look for the printer and the output queue to use.

The system looks, in the user profile, at the parameters of output queue and printer device.

Output queue: In the example user profile, you can see the output queue value (item **2**) specified is *WRKSTN. This tells the system to use the value specified in the output queue (OUTQ) parameter of the *workstation description* as the output queue name.

```
Display User Profile - *BASIC
User profile  . . . . . . . . . . . . . :     LAWSON
Previous sign-on  . . . . . . . . . . . :     04/15/91   13:47:07
Sign-on attempts not valid  . . . . . . :     0
Date password last changed  . . . . . . :     04/09/91
Password expiration interval  . . . . . :     *SYSVAL
  Date password expires . . . . . . . . :     05/09/91
Set password to expired . . . . . . . . :     *NO
User class  . . . . . . . . . . . . . . :     *PGMR
Special authority . . . . . . . . . . . :     *ALLOBJ
*JOBCTL
*SAVSYS
*SECADM
*SPLCTL
Group profile . . . . . . . . . . . . . :     *NONE
Owner . . . . . . . . . . . . . . . . . :     *USRPRF
Group authority . . . . . . . . . . . . :     *NONE
Assistance level. . . . . . . . . . . . :     *INTERMED
Current library . . . . . . . . . . . . :     *CRTDFT
Initial menu  . . . . . . . . . . . . . :     MAIN
  Library . . . . . . . . . . . . . . . :        *LIBL
Initial program . . . . . . . . . . . . :     *NONE
```

```
Library . . . . . . . . . . . . . . . :
Limit capabilities . . . . . . . . . . . :        *NO
Text  . . . . . . . . . . . . . . . . . :
pat lawson   534/030-2    3-6051
Display sign-on information . . . . . . :          *SYSVAL
Limit device sessions . . . . . . . . . :          *SYSVAL
Keyboard buffering. . . . . . . . . . . :          *SYSVAL
Maximum storage allowed . . . . . . . . :          *NOMAX
Storage used  . . . . . . . . . . . . . :          519
Highest scheduling priority . . . . . . :          3
**************************************************************************
1  Job description . . . . . . . . . . :          QDFTJOBD
**************************************************************************
   Library . . . . . . . . . . . . . . :          QGPL
Accounting code . . . . . . . . . . . . :
Message queue . . . . . . . . . . . . . :          LAWSON
   Library . . . . . . . . . . . . . . :          QUSRSYS
Message queue delivery  . . . . . . . . :          *NOTIFY
Message queue severity  . . . . . . . . :          0
**************************************************************************
2  Output queue  . . . . . . . . . . . :          *WRKSTN
   Library . . . . . . . . . . . . . . :
**************************************************************************
3  Printer device  . . . . . . . . . . :          *WRKSTN
**************************************************************************
Special Environment . . . . . . . . . . :          *SYSVAL
Attention program . . . . . . . . . . . :          *NONE
   Library . . . . . . . . . . . . . . :
User options  . . . . . . . . . . . . . :          *NONE
```

Understanding the workstation description

On the following page is an example of a **workstation description** with the values
set to the system-supplied defaults.

Each user signs on to the system at a display station. Many display stations can be
attached to an iSeries server. Each display station that is attached to the system has
a **workstation description**.

The workstation description, for a display station, is a collection of information
that tells the system how the display station is to be used.

From a printing standpoint, the parameters labeled **1** Printer device and **2**
Output queue are very important.

When you sign on to the system, if no defaults or system values have been
changed, your output will be sent to the output queue and the printer device
specified in the workstation description of the display station where you are
signed on.

Note: Any batch job submitted from the interactive job would use the same printer
device or output queue that is currently specified in the workstation
description.

Output queue: In the example workstation description, you can see the output
queue value (item **2**) specified is *DEV. This tells the system to use the value
specified in the printer device parameter of the printer file as the output queue
name.

In our example, the value in the PRTDEV parameter of the job description is
*USRPRF, which tells the system to look at the PRTDEV parameter in the user

profile. The value in that parameter is *WRKSTN, which tells the system to look at the PRTDEV parameter in the workstation description.

Printer device: In the example workstation description, you can see the printer device value (item **1**) specified is *SYSVAL. This tells the system to look at the system value QPRTDEV. This IBM-supplied system value has a printer assigned to it. For example: PRT01 could be the value assigned to the system value QPRTDEV.

To find out what printer is assigned to QPRTDEV, see "Understanding system values" on page 22.

```
Display Device Description                    MSP38360

Device description . . . . . . . . :    DSP10
Option . . . . . . . . . . . . . . :    *BASIC
Category of device . . . . . . . . :    *DSP
Device class . . . . . . . . . . . :    *LCL

Device type  . . . . . . . . . . . :    5291
Device model . . . . . . . . . . . :    2
Port number  . . . . . . . . . . . :    6
Switch setting . . . . . . . . . . :    6
Online at IPL  . . . . . . . . . . :    *YES
Attached controller  . . . . . . . :    CTL02
Keyboard language type . . . . . . :    USB
Character identifier . . . . . . . :    *KBDTYPE
Allow blinking cursor  . . . . . . :    *YES
*************************************************************************
1  Printer device  . . . . . . . . . :       *SYSVAL
2  Output queue  . . . . . . . . . . :       *DEV
Library . . . . . . . . . . . . . . :
*************************************************************************
Printer file . . . . . . . . . . . :    QSYSPRT
Library  . . . . . . . . . . . . . :    *LIBL
Text . . . . . . . . . . . . . . . :    CREATED BY AUTO-CONFIGURATION
```

Stage 2: Job run time

At job run time, the values in the printer file and the job attributes are used. For additional information about how the OUTQ and PRTDEV job attributes can be changed, see the Work Management topic in the iSeries Information Center.

Understanding printer files

Notes:

1. The information in this chapter on printer files is quite brief and intended only to introduce the printer file concept to you. For more detailed information on printer file support go to Chapter 2, "Printer file support" on page 41.
2. Also, for this discussion, assume that the printer file parameter SPOOL is set to *YES. This means that spooled files will be sent to an output queue (OUTQ).

As you learned earlier in this chapter, printer files contain many parameters that describe how the system is to operate on the data as it passes between your application program and the printer. One of these parameters, the output queue (OUTQ) parameter, tells the system which output queue receives your spooled files.

On the following page is a diagram of the system-supplied printer file called QSYSPRT. Locate the spooled output queue parameter (**1**) on that diagram.

You might expect that the location of your printed output can be determined by knowing the name of the spooled output queue specified in your printer file. However, in the default printer file QSYSPRT, the value specified is *JOB.

This means that the QSYSPRT printer file tells the system to look at your *job attributes* to determine the name of the output queue (OUTQ).

For more detailed information about job attributes, see the Work Management topic in the iSeries Information Center.

```
                        Display Spooled File
File  . . . . . :   QPDSPFD                    Page/Line   1/1
Control . . . . .                              Columns     1 - 78
Find  . . . . . .
2/09/98                 Display File Description

DSPFD Command Input
 File . . . . . . . . . . . . . . . . . . . : FILE       QSYSPRT
   Library . . . . . . . . . . . . . . . . . :            *LIBL
 Type of information . . . . . . . . . . . . : TYPE       *ALL
 File attributes . . . . . . . . . . . . . . : FILEATR    *ALL
 System  . . . . . . . . . . . . . . . . . . : SYSTEM     *LCL
 File Description Header
 File  . . . . . . . . . . . . . . . . . . . : FILE       QSYSPRT
 Library . . . . . . . . . . . . . . . . . . :            QSYS
 Type of file  . . . . . . . . . . . . . . . :            Device
 Device type . . . . . . . . . . . . . . . . :            Printer
 Auxiliary storage pool ID . . . . . . . . . :            01
 Device File Attributes
 Externally described file . . . . . . . . . :            No
 File level identifier . . . . . . . . . . . :            0980109013920
 Creation date . . . . . . . . . . . . . . . :            01/09/98
 Text 'description'  . . . . . . . . . . . . : TEXT       System non-describ
Spool the data  . . . . . . . . . . . . . . : SPOOL      *YES
Maximum devices . . . . . . . . . . . . . . :            1
 User specified DBCS data  . . . . . . . . . : IGCDTA     *NO
 Maximum file wait time  . . . . . . . . . . : WAITFILE   *IMMED
 Share open data path  . . . . . . . . . . . : SHARE      *NO
 Record format level check . . . . . . . . . : LVLCHK     *NO
 Number of record formats  . . . . . . . . . :            1
 User buffer length  . . . . . . . . . . . . :            0
 Number of devices . . . . . . . . . . . . . :            1
 Separate indicator area . . . . . . . . . . : INDARA     No
 Coded character set identifier  . . . . . . : CCSID         0
Printer Attributes
 Device  . . . . . . . . . . . . . . . . . . : DEV        *JOB
 Printer device type . . . . . . . . . . . . : DEVTYPE    *SCS
 Page size                                     PAGESIZE
   Length  . . . . . . . . . . . . . . . . . :               34
   Width . . . . . . . . . . . . . . . . . . :               132
   Measurement Method  . . . . . . . . . . . :            *ROWCOL
 Lines per inch  . . . . . . . . . . . . . . : LPI        6
 Characters per inch . . . . . . . . . . . . : CPI        10
 Front margin  . . . . . . . . . . . . . . . : FRONTMGN   *DEVD
 Back margin . . . . . . . . . . . . . . . . : BACKMGN    *FRONTMGN
 Overflow line number  . . . . . . . . . . . : OVRFLW     30
 Fold records  . . . . . . . . . . . . . . . : FOLD       *NO
 Degree of page rotation . . . . . . . . . . : PAGRTT     *AUTO
 Hardware justification  . . . . . . . . . . : JUSTIFY     0
 Print on both sides . . . . . . . . . . . . : DUPLEX     *NO
 Defer Write . . . . . . . . . . . . . . . . : DFRWRT     *YES
 Unprintable character action                  RPLUNPRT
   Replace character . . . . . . . . . . . . :            *YES
   Replacement character . . . . . . . . . . :            ' '   X'40'
 Print text  . . . . . . . . . . . . . . . . : PRTTXT     *JOB
 Align page  . . . . . . . . . . . . . . . . : ALIGN      *NO
```

```
Control character . . . . . . . . . . . . . . : CTLCHAR    *NONE
Channel values  . . . . . . . . . . . . . . : CHLVAL     *NORMAL
Fidelity . . . . . . . . . . . . . . . . . : FIDELITY   *CONTENT
Printer quality . . . . . . . . . . . . . . : PRTQLTY    *STD
Form feed . . . . . . . . . . . . . . . . . : FORMFEED   *DEVD
Source drawer . . . . . . . . . . . . . . . : DRAWER     1
Output bin  . . . . . . . . . . . . . . . . : OUTBIN     *DEVD
Font                                          FONT
   Identifier  . . . . . . . . . . . . . . :            *CPI
   Point size  . . . . . . . . . . . . . . :            *NONE
Character identifier  . . . . . . . . . . . : CHRID      *CHRIDCTL
Decimal format  . . . . . . . . . . . . . . : DECFMT     *JOB
Font character set  . . . . . . . . . . . . : FNTCHRSET  *FONT
Coded font . . . . . . . . . . . . . . . . : CDEFNT     *FNTCHRSET
Table Reference Characters  . . . . . . . . : TBLREFCHR  *NO
AFP Chars . . . . . . . . . . . . . . . . . : AFPCHARS   *NONE
Page definition . . . . . . . . . . . . . . : PAGDFN     *NONE
Form definition . . . . . . . . . . . . . . : FORMDF     *NONE
Form type . . . . . . . . . . . . . . . . . : FORMTYPE   *STD
Pages per side  . . . . . . . . . . . . . . : MULTIUP    1
Reduce output . . . . . . . . . . . . . . . : REDUCE     *TEXT
Unit of measure . . . . . . . . . . . . . . : UOM        *INCH
Front side overlay  . . . . . . . . . . . . : FRONTOVL   *NONE
Back side overlay . . . . . . . . . . . . . : BACKOVL    *FRONTOVL
IPDS pass through . . . . . . . . . . . . . : IPDSPASTHR *DEVD
Convert line data . . . . . . . . . . . . . : CVTLINDTA  *NO
User resource library list  . . . . . . . . : USRRSCLIBL *DEVD
Corner staple . . . . . . . . . . . . . . . : CORNERSTPL *NONE
Edge stitch  . . . . . . . . . . . . . . . : EDGESTITCH
 Reference edge . . . . . . . . . . . . . . :            *NONE
Saddle stitch . . . . . . . . . . . . . . . : SADLSTITCH
 Reference edge . . . . . . . . . . . . . . :            *NONE
Font resolution   . . . . . . . . . . . . . : FNTRSL     *DEVD
DBCS extension characters . . . . . . . . . : IGCEXNCHR  *YES
DBCS character rotation . . . . . . . . . . : IGCCHRRTT  *NO
DBCS characters per inch  . . . . . . . . . : IGCCPI     *CPI
DBCS SO/SI spacing  . . . . . . . . . . . . : IGCSOSI    *YES
DBCS Coded font . . . . . . . . . . . . . . : IGCCDEFNT  *SYSVAL
Spooling Description
 ********************************************************************************
1    Spooled output queue  . . .    : OUTQ      *JOB
 ********************************************************************************
Max spooled output records  . . . . . . . . : MAXRCDS          100000
Spooled output schedule . . . . . . . . . . : SCHEDULE   *FILEEND
Copies  . . . . . . . . . . . . . . . . . . : COPIES     1
Page range to print                           PAGERANGE
   Starting page . . . . . . . . . . . . . . :                 1
   Ending page . . . . . . . . . . . . . . . :            *END
File separators . . . . . . . . . . . . . . : FILESEP    0
Hold spooled file . . . . . . . . . . . . . : HOLD       *NO
Save spooled file . . . . . . . . . . . . . : SAVE       *NO
Output priority (on OUTQ) . . . . . . . . . : OUTPTY     *JOB
User data . . . . . . . . . . . . . . . . . : USRDTA     *SOURCE
Spool file owner  . . . . . . . . . . . . . : SPLFOWN    *CURUSRPRF
User defined option . . . . . . . . . . . . : USRDFNOPT  *NONE
User defined data . . . . . . . . . . . . . : USRDFNDTA  *NONE
User defined object                           USRDFNOBJ
   Object  . . . . . . . . . . . . . . . . . :            *NONE
     Library . . . . . . . . . . . . . . . . :
   Object type . . . . . . . . . . . . . . . :
Record Format List

                 Record  Format Level
 Format      Fields  Length  Identifier
 QSYSPRT          0       0  0000000000000
   Text . . . . . . . . . . . . . . . . . . :
```

```
Total number of formats  . . . . . . . . . . :        1
Total number of fields . . . . . . . . . . . :        0
Total record length  . . . . . . . . . . . . :        0
```

Understanding system values

System values are objects supplied by IBM and shipped with the system. System values control such things as system date, system time, system console (display station), system printer, and so on. The system value most important to printing is the system printer.

The name of the system value that specifies the name of the default system printer is **QPRTDEV**. The value supplied by IBM for the system value QPRTDEV is PRT01.

Displaying the QPRTDEV system value: As a user of the system, you can display any of the IBM-supplied system values.

To display the QPRTDEV system value, type DSPSYSVAL QPRTDEV and press the Enter key.

If the value supplied by IBM has not been changed, the name of the system printer is PRT01.

Where did the output go?

Since the printer file in the example specified SPOOL = *YES, the application program sent a spooled file to an output queue. The system looked at the printing elements in the order shown in "Understanding the hierarchy of the printing elements" on page 15. In the example, the spooled file was sent to the output queue with the same name as the system printer, PRT01.

How the printing elements control print activity

This chapter has discussed the many elements that control where your requests for printed output go.

If you are a programmer or system operator, the flexibility of creating additional output queues can help you manage and balance your printing work load.

If you are the person responsible for adding new users to the system, you can avoid confusion by specifying the necessary print device values you want in the user's *user profile*.

Go to "Examples: where your printing would go" on page 23 for some examples that show how the printing elements work with each other.

Other elements that control print activity

Prior to V4R4, storing the spooled file under a QPRTJOB could influence where a spooled file would be logically stored. If a printer file had *JOB specified in the output queue parameter, the output queue parameter in the QPRTJOB would be used. The value in the output queue parameter in current job would be ignored. Beginning with V4R4, the output queue of the current job controls where the spooled files are stored.

You can do one or more of the following to affect your print activity:

- You can use the spooled file owner parameter to store the printed output under a different job and user.

- You can use the QWTSETP API, and switch to a different user profile. See "Working with a QPRTJOB" on page 157, for more information.
- You can create a data area to use the output queue value from the QPRTJOB job by doing the following: Create a data area that is called QPRTJOB. The data area must exist in either QUSRSYS or the first product library in the library list of the current job. The data area must be of type logical and have a value of false (0). The user profile QSPL must own the data area. If the QPRTJOB data area is not found, or there are any problems with it, the output queue value of the current job will be used instead. Problems with the data area can include: damage to the data area, not the correct type, not owned by user QSPL, or has a logical value of true ('1').
- You can use the CL command, ADDENVVAR, to trigger the OS to add an entry in the DATQ for each spool file created. For more information about creating data queues, see "Environment variable QIBM_NOTIFY_CRTSPLF data queue support" on page 132

Examples: where your printing would go

The following examples show where your printed output would go based on certain values that are assigned to the various printing elements. The graphics indicate the values assigned to the job before the job begins or before the spool file gets created.

Example 1: Determine your output queue

In the printer file, assume that:
- The spooled file owner is *CURUSRPRF.
- Output queue value is *JOB
- Printer device value is *JOB
- SPOOL = *YES

  Because SPOOL = *YES, the output must go to an output queue.

Also assume that there has not been a switch to an alternate user profile.

At job start time (initiation)

Job description	User profile	Workstation description		Job attributes
OUTQ(*USRPRF)	OUTQ(*WRKSTN)	OUTQ(*DEV)		OUTQ (*DEV)
PRTDEV(*USRPRF)	PRTDEV(*WRKSTN)	PRTDEV(*SYSVAL)		PRTDEV (PRT01)

System value   QPRTDEV(PRT01)

At job runtime (spooled file creation time)

Printer file	Job attributes	Output queue
OUTQ(*JOB)	OUTQ(*DEV)	PRT01
DEV(*JOB)	PRTDEV(PRT01)	

RBAFT510-0
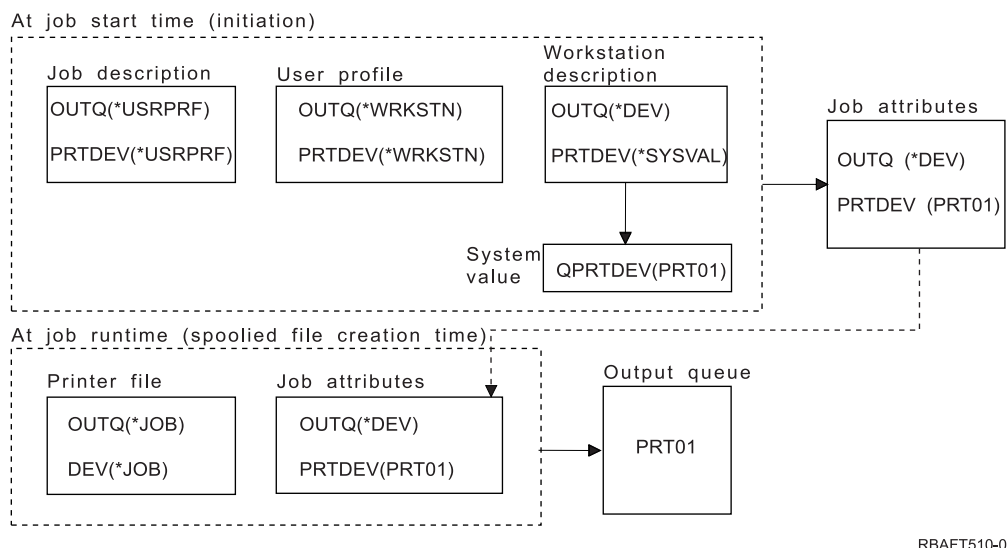
In example 1, at job initiation, the following takes place:

The system looks at the OUTQ parameter in the job description, that value is *USRPRF. This tells the system to look at the OUTQ parameter in the user profile. In this example, that value is *WRKSTN. This tells the system to look at the OUQ parameter in the workstation description. In the workstation description, the OUTQ parameter value is *DEV. *DEV is stored in the job attribute OUTQ.

The system looks a the PRTDEV parameter in the job description, that value is *USRPRF. This tells the system to look at the PRTDEV parameter in the user profile. In this example, that value is *WRKSTN. This tells the system to look at the PRTDEV parameter in the workstation description. This tells the system to look at hte system value QPRTDEV and to use the output queue on the system that has the same name as the system printer named in the system value QPRTDEV. In this example, that is PRT01. PRT01 is stored in the job attribute PRTDEV.

At job run time, the following takes place:

When a spooled file is to be created, the system looks at the OUTQ parameter in the printer file for the output queue name. In this example, that value is *JOB. This tells the system to look at the OUTQ attribute for the job. The OUTQ job attribute was set to *DEV during the job initiation stage. The OUTQ job attribute, *DEV, tells the system to look at the DEV parameter in the printer file. The value in the DEV parameter of the printer file is *JOB. That value tells the system to look at the PRTDEV attribute of the job. If the IBM-supplied value for QPRTDEV has not been changed, the printer device name is PRT01 and the output queue name is PRT01.

If the IBM-supplied value for QPRTDEV has not been modified, the printer device name is PRT01 and the output queue name is PRT01.

Example 2: Determine your output queue

In the printer file, assume that:
- Spooled file owner is *CURUSRPRF.
- Output queue value is PRT04
- Printer device value is PRT08
- SPOOL = *YES

At job start time

Job description	User profile	Workstation description	Job attributes
OUTQ(PRT03) PRTDEV(PRT03)	OUTQ(PRT02) PRTDEV(PRT02)	OUTQ(OUTQ7) PRTDEV(PRT07)	OUTQ (PRT03) PRTDEV (PRT03)

At job runtime

Printer file	Job attributes	Output queue
OUTQ(PRT04) DEV(PRT08)	OUTQ(PRT03) PRTDEV(PRT03)	PRT04
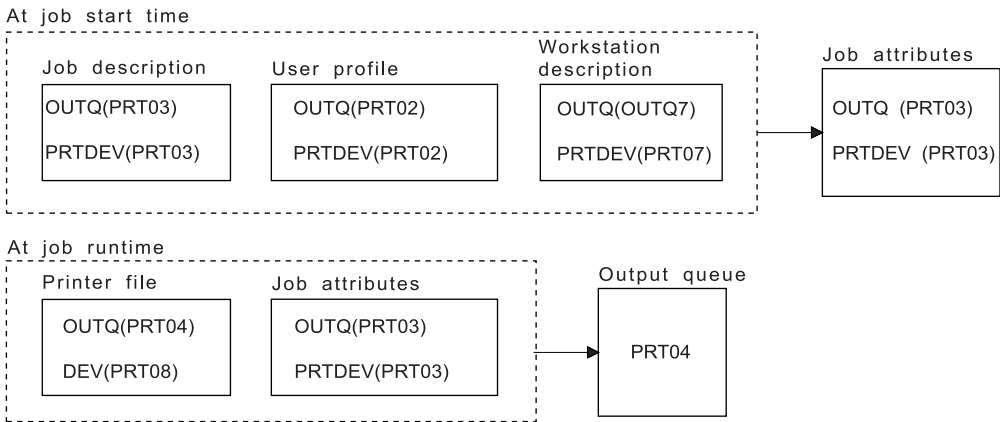
RBAFT504-1

In example 2, the output queue would be PRT04. The system found the output queue parameter value of PRT04 in the printer file rather than a value that would point it to your OUTQ job attribute.

Example 3: Determine your output queue

In the printer file, assume the following:

- Output queue value is *JOB
- Printer device value is *JOB
- SPOOL = *YES
- Spooled file owner is *CURGRPPRF

Also assume:

- The job did not switch to an alternate user profile.
- The current user has a group profile GRP.

Job initiation

```
Job description          User profile          Workstation
                                               description
                                                                    Job attributes
OUTQ(*USRPRF)            OUTQ(*WRKSTN)         OUTQ(*DEV)
                                                                    OUTQ (*DEV)
PRTDEV(*USRPRF)          PRTDEV(*WRKSTN)       PRTDEV(*SYSVAL)
                                                                    PRTDEV (PRT01)
xxxxxx/GRP/QPRTJOB
Job attributes
                                            System
OUTQ(*DEV)                                   value    QPRTDEV(PRT01)
```

At job runtime

```
Printer file            Job attributes          Output queue

OUTQ(*JOB)              OUTQ(*DEV)
DEV(*JOB)                                         PRT01
SPLFOWN                 PRTDEV(PRT01)
(*CURGRPPRF)
```

RBAFT515-1

Note: Because the SPLFOWN parameter of the printer file is *CURGRPPRF, the spooled file will be created under job xxxxxx/GRP/QPRTJOB (where xxxxxx is 000000–999999). For more information on the SPLFOWN parameter see the CL Reference topic in the iSeries Information Center.

In example 3, at job initiation, the following takes place:

The system looks at the OUTQ value in the current job description. The value *USRPRF, in the job description, tells the system to look at the OUTQ parameter in the user profile. The value in the OUTQ parameter of the user profile is *WRKSTN. This tells the system to look at the OUTQ parameter in the workstation description. In the workstation description, the OUTQ parameter is *DEV. In the job attributes, the OUTQ job attribute gets set to *DEV.

The system looks PRTDEV parameter in the job description. The value *USRPRF, in the job description, tells the system to look at the PRTDEV parameter in the user profile. The value *WRKSTN, in the user profile, tells the system to look at the PRTDEV parameter in the workstation description. The value *SYSVAL, in the workstation description, tells the system to look at the system value and use the value set for QPRTDEV. The value in QPRTDEV is PRT01. PRT01 becomes the value for the PRTDEV job attribute.

At job runtime the following takes place:

The sytem looks at the OUTQ value in the printer file. That value *JOB tells the system to use the OUTQ job attribute for job xxxxxx/GRP/QPRTJOB which is *DEV. This tells the system to look at the DEV attribute in the printer file, which is *JOB. The value, *JOB tells the system to look at the PRTDEV value in the job attributes. The value for the PRTDEV job attribute is PRT01.

Example 4: Determine your output queue

In the printer file, assume the following:
* Output queue value is *JOB
* Printer device value is *JOB
* SPOOL = *YES
* Spooled file owner is *CURUSRPRF

Also assume that there has been a switch to an alternate user profile USR.
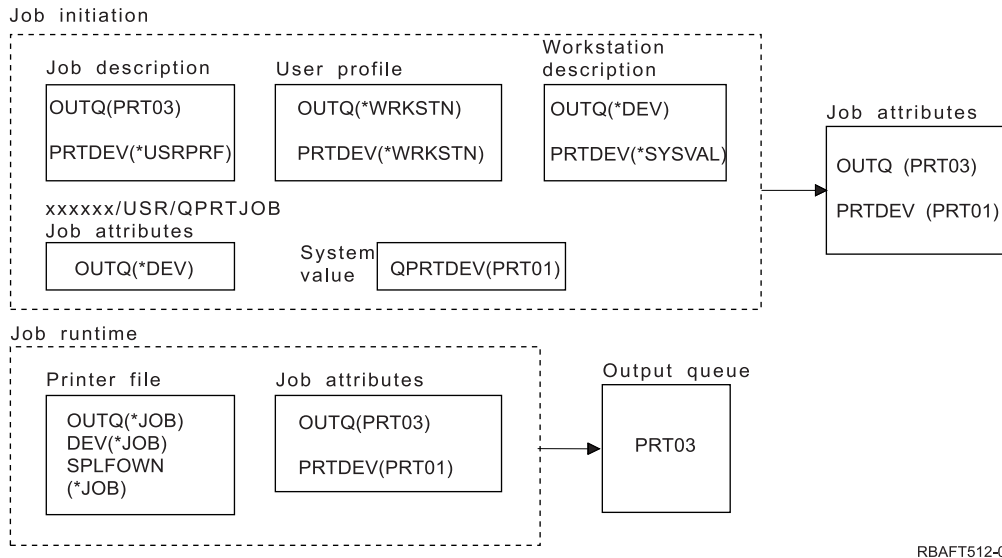

```
Job initiation

     Job description        User profile          Workstation
                                                  description
     OUTQ(PRT03)            OUTQ(*WRKSTN)          OUTQ(*DEV)             Job attributes

     PRTDEV(*USRPRF)        PRTDEV(*WRKSTN)        PRTDEV(*SYSVAL)        OUTQ (PRT03)

     xxxxxx/USR/QPRTJOB                                                  PRTDEV (PRT01)
     Job attributes
        OUTQ(*DEV)          System    QPRTDEV(PRT01)
                            value

Job runtime
                                                  Output queue
     Printer file           Job attributes
        OUTQ(*JOB)             OUTQ(PRT03)
        DEV(*JOB)                                    PRT03
        SPLFOWN                PRTDEV(PRT01)
        (*JOB)
                                                              RBAFT512-0
```

Note: The SPLFOWN parameter of the printer file is *JOB, and the job has switched to user profile USR. The current job will create the spooled file. For more information on the SPLFOWN parameter see the CL Reference topic in the iSeries Information Center.

The system looks at the OUTQ parameter in the printer file for the output queue name. The value, in this example, *JOB, tells the system to look at the OUTQ job attribute. Because the SPFLOWN parameter is set to *JOB, the job attribute OUTQ of the current job is used. The value is PRT03. In this example, the spooled file goes to output queue PRT03.

Example 5: Determine your output queue

In the printer file, assume the following:
* Output queue value is *JOB
* Printer device value is *JOB
* SPOOL = *YES
* Spooled file owner is *CURUSRPRF

Also assume:

- There has been a switch to an alternate user profile USR.
- The data area QPRTJOB, of type *LGL, with a value of false (0), exists in library QUSRSYS and is owned by QSPL user profile.
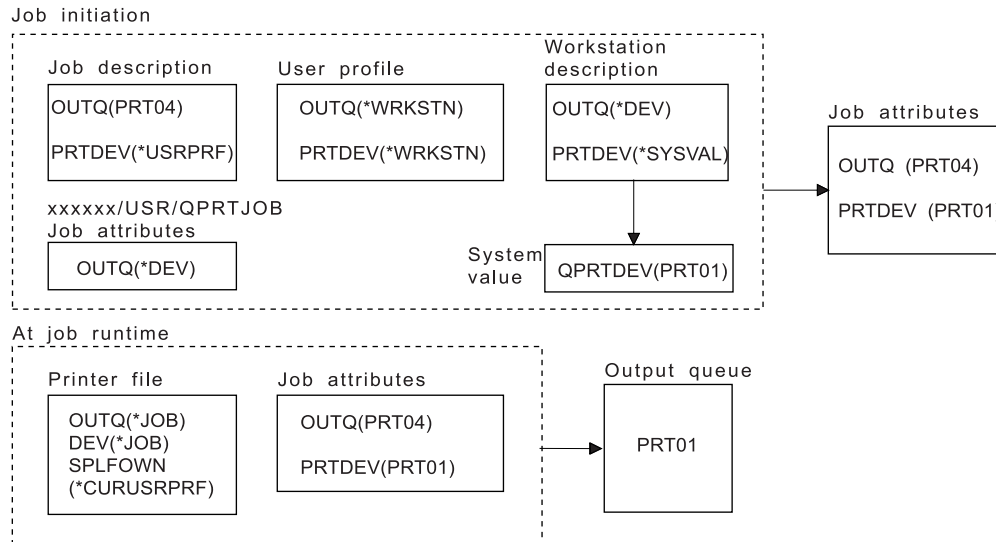
```
Job initiation
┌──────────────────────────────────────────────────────────────────────────┐
│                                       Workstation                          │
│  Job description     User profile      description                         │
│  ┌──────────────┐   ┌──────────────┐  ┌──────────────┐    Job attributes   │
│  │OUTQ(PRT04)   │   │OUTQ(*WRKSTN) │  │OUTQ(*DEV)    │   ┌──────────────┐   │
│  │              │   │              │  │              │   │OUTQ (PRT04)  │   │
│  │PRTDEV(*USRPRF)│   │PRTDEV(*WRKSTN)│  │PRTDEV(*SYSVAL)│   │              │   │
│  └──────────────┘   └──────────────┘  └──────────────┘   │PRTDEV (PRT01)│   │
│  xxxxxx/USR/QPRTJOB                           │          └──────────────┘   │
│  Job attributes                               ▼                            │
│  ┌──────────────┐               System ┌──────────────┐                    │
│  │OUTQ(*DEV)    │               value  │QPRTDEV(PRT01) │                    │
│  └──────────────┘                      └──────────────┘                    │
└──────────────────────────────────────────────────────────────────────────┘
At job runtime
┌──────────────────────────────────────────────────┐   Output queue
│  Printer file       Job attributes                │   ┌──────────────┐
│  ┌──────────────┐   ┌──────────────┐              │   │              │
│  │OUTQ(*JOB)    │   │OUTQ(PRT04)   │              │   │  PRT01       │
│  │DEV(*JOB)     │   │              │ ─────────────┼──▶│              │
│  │SPLFOWN       │   │PRTDEV(PRT01) │              │   │              │
│  │(*CURUSRPRF)  │   └──────────────┘              │   └──────────────┘
│  └──────────────┘                                 │
└──────────────────────────────────────────────────┘
                                           RBAFT513-0
```

Note: The SPLFOWN parameter of the printer file is *CURUSRPRF, and the job has switched to user profile USR. The spooled file will be created under job xxxxxx/USR/QPRTJOB (where xxxxxx is 000000–999999). For more information on the SPLFOWN parameter see the CL Reference topic in the iSeries Information Center.

In example 5, at job initiation, the following takes place:

The system looks at the OUTQ parameter in the job description. That value, PRT04 tells the system that it does not need to look any further and sets the OUTQ job attribute to PRT04.

The value *USRPRF in the PRTDEV parameter of the job description, tells the system to look at the PRTDEV attribute of the user profile. In the user profile, the value *WRKSTN tells the system to look at the PRTDEV parameter in the workstation description. That value, *SYSVAL tells the system to look at the system value QPRTDEV and to use the output queue that is named in that value. In this example, the value is PRT01 and it is stored in the job attribute PRTDEV.

At job run time, the following takes place:

The system looks at the OUTQ parameter in the printer file for the output queue name. That value *JOB, tells the system to look at the OUTQ attribute of the job. There is a data area, QPRTJOB in QUSRSYS, that is owned by the user profile QSPL that has a logical value of false. Because of this data area, the system will look at the OUTQ attribute for job xxxxxx/USR/QPRTJOB. In the xxxxxx/USR/QPRTJOB, the OUTQ attribute value *DEV tells the system to look at the DEV parameter in the printer file. The value *JOB in the DEV parameter of the printer file, tells the system to look at the PRTDEV attribute of the current job. That value is PRT01.

Example 6: Determine your output queue

In the printer file, assume the following:

- Output queue value is *JOB
- Printer device value is *JOB
- SPOOL = *YES
- Spooled file owner is *JOB

Also assume that there has been a switch to an alternate user profile USR.

Job initiation

Job description
OUTQ(PRT03)
PRTDEV(*USRPRF)

User profile
OUTQ(*WRKSTN)
PRTDEV(*WRKSTN)

Workstation description
OUTQ(*DEV)
PRTDEV(*SYSVAL)

Job attributes
OUTQ (PRT03)
PRTDEV (PRT01)

xxxxxx/USR/QPRTJOB
Job attributes
OUTQ(*DEV)

System value   QPRTDEV(PRT01)

At job runtime

Printer file
OUTQ(*JOB)
DEV(*JOB)
SPLFOWN
(*CURUSRPRF)

Job attributes
OUTQ(PRT03)
PRTDEV(PRT01)

Output queue
PRT03
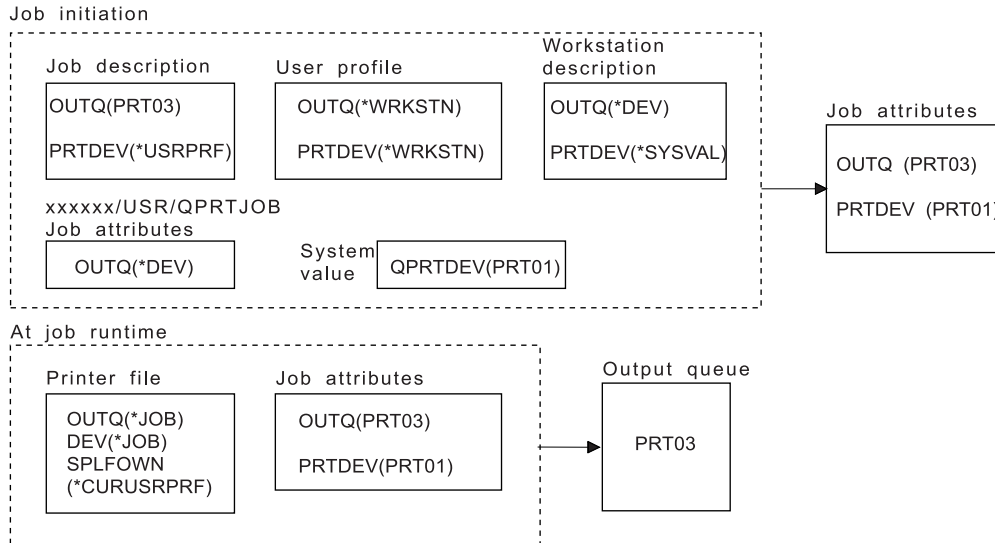
RBAFT514-0

Note: The SPLFOWN parameter of the printer file is *CURUSRPRF, and the job has made a switch to user profile USR. The spooled file will be created under job xxxxxx/USR/QPRTJOB (where xxxxxx is 000000–999999). For more information on the SPLFOWN parameter see the CL Reference topic in the iSeries Information Center.

The system looks at the OUTQ parameter in the printer file for the output queue name. That value, *JOB, tells the system to look at the OUTQ job attribute. The system will look at the OUTQ job attribute of the current job, which is PRT03.

Example 7: Determine your output queue

In the printer file, assume the following:

- Output queue value is *JOB
- Printer device value is *JOB
- SPOOL = *YES
- Spooled file owner is *CURUSRPRF

Also assume:

- There has been a switch to alternate user profile USR.
- The current user has a group profile X.
- The data area QPRTJOB, of type *LGL, with a value of false (0), exists in library QUSRSYS and is owned by QSPL user profile.

- Another data area QPRTJOB, of type *LGL, with a value of true (1), exists in the first product library of the current job's library list. The QSPL user profile owns the data area.

```
Job initiation
  Job description          User profile             Workstation
                                                    description
  OUTQ(PRT04)              OUTQ(*WRKSTN)            OUTQ(*DEV)           Job attributes

  PRTDEV(*USRPRF)          PRTDEV(*WRKSTN)          PRTDEV(*SYSVAL)      OUTQ (PRT04)

  xxxxxx/USR/QPRTJOB                                                    PRTDEV (PRT01)
  Job attributes
     OUTQ(*DEV)            System    QPRTDEV(PRT01)
                           value

Job runtime
  Printer file             Job attributes           Output queue
     OUTQ(*JOB)               OUTQ(PRT04)
     DEV(*JOB)
     SPLFOWN                  PRTDEV(PRT01)             PRT04
     (*CURUSRPRF)

                                                       RBAFT511-0
```

Note: The SPLFOWN parameter of the printer file is *CURUSRPRF, and the job has switched to alternate user profile USR. The spooled file will be created under job xxxxxx/USR/QPRTJOB (where xxxxxx is 000000–999999).

The system looks at the OUTQ parameter in the printer file for the output queue name. That value, *JOB, tells the system to look at the OUTQ job attribute. Because data area QPRTJOB with a logical value of true exists, the system will look at the OUTQ job attribute of in the current job, which is PRT04.

Example 8: Determine Your Printer Name

In the printer file, assume that:
- SPOOL = *NO
- The spooled file owner is *CURUSRPRF.
- Output queue value is *JOB
- Printer device value is *JOB

Also assume that there has not been a switch to an alternate user profile.

Job initiation

Job description	User profile	Workstation description		Job attributes
OUTQ(*USRPRF)	OUTQ(OUTQ1)	OUTQ(OUTQ5)	→	OUTQ (OUTQ1)
PRTDEV(*USRPRF)	PRTDEV(PRT02)	PRTDEV(PRT05)		PRTDEV (PRT02)

Job runtime

Printer file	Job attributes		Printer device
OUTQ(*JOB)	OUTQ(OUTQ1)	→	PRT02
DEV(*JOB)	PRTDEV(PRT02)		
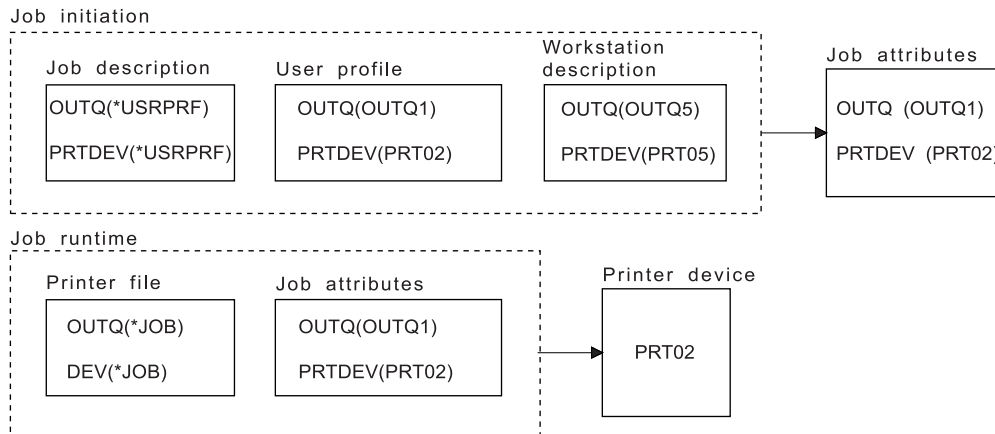
RBAFT505-1

In example 8, at job initiation, the following takes place:

The system looks at the OUTQ parameter in the job description. That value *USRPRF tells the system to look at the OUTQ parameter in the user profile. The value of the OUTQ parameter in the user profile is OUTQ1. Because this is the name of a specific output queue, this value is stored as the OUTQ value in the job attributes.

The system looks at the PRTDEV parameter in the job description. That value *USRPRF tells the system to look at the PRTDEV parameter in the user profile. The PRTDEV value in the user profile is PRT02. Because this is the name of a specific printer device, the system stops looking and stores this value as the PRTDEV value in the job attributes.

At job run time the following takes place:

In this example, your printer device would be PRT02. This is because the system looked first into the printer file and found the PRTDEV parameter value to be *JOB, which sent it to the job attribute PRTDEV.

The job attribute PRTDEV value is PRT02.

In this example, the printer file specified SPOOL = *NO. Your output would go directly to PRT02 for printing and no output queue would be used.

Example 9: Determine Your Printer Name

In the printer file, assume that:
- SPOOL = *NO.
- The spooled file owner is *CURUSRPRF.
- Output queue value is *JOB
- Printer device value is *JOB

Also assume that there has not been a switch to an alternate user profile.

Job initiation

Job description	User profile	Workstation description		Job attributes
OUTQ(*USRPRF)	OUTQ(*WRKSTN)	OUTQ(PRT02)	→	OUTQ (PRT02)
PRTDEV(*USRPRF)	PRTDEV(*WRKSTN)	PRTDEV(PRT03)		PRTDEV (PRT03)

Job runtime

Printer file	Job attributes		Printer device
OUTQ(*JOB)	OUTQ(PRT02)	→	PRT03
PRTDEV(*JOB)	PRTDEV(PRT03)		
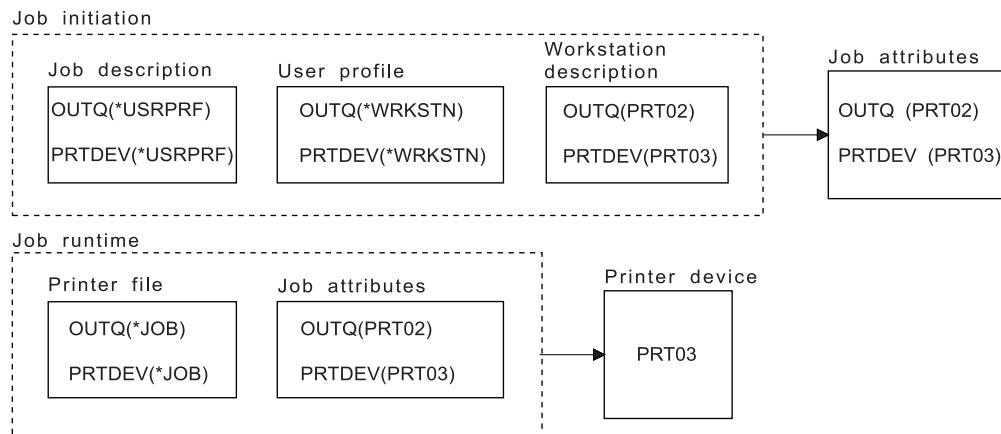
RBAFT503-1

In this example, at job initiation, the following takes place:

The system looks at the OUTQ parameter in the job description. That value, *USRPRF tells the system to look at the OUTQ parameter in the user profile. In the user profile, the OUTQ parameter value is *WRKSTN. This value tells the system to look at the OUTQ parameter in the workstation description. The OUTQ value in the workstation description is PRT02. This value is stored as the OUTQ value in the job attributes.

The system looks at the PRTDEV value in the job description. That PRTDEV value in the job description is *USRPRF. That value tells the system to look at the PRTDEV value in the user profile. The PRTDEV value in the user profile is *WRKSTN. That value tells the system to look at the PRTDEV value in the workstation description. The PRTDEV value in the workstation description is PRT03. This value is stored as the PRTDEV value in ithe job attributes.

At job run time, the following takes place:

The system looked at the printer file and found the PRTDEV parameter value to be *JOB, which tells the system to look next in the job printer attribute PRTDEV.

In this example, that value is PRT03.

In this example, the printer file specified SPOOL = *NO. Your output would go directly to PRT03 for printing and no output queue would be used.

REMEMBER:

> You must know the value (*YES or *NO) of the SPOOL parameter in the printer file to determine if your output goes to an output queue or to a printer. If SPOOL = *YES, a spooled file goes to an output queue. If SPOOL = *NO, the output goes directly to a printer.

Printing in a batch environment

When a user signs on to the iSeries server, an output queue (OUTQ) and a printer device (PRTDEV) are established for that job.

The names for the output queue and printer device are resolved when the system searches through the user profile, job description, workstation description, and system values.

How output is routed in a batch job

A similar process occurs when batch jobs are started on the system, with two differences:

- The output queue and printer device values are passed to the batch job from the job that started it. This means that no resolution of the output queue and printer device needs to be done.
- The value *WRKSTN has no real meaning because batch jobs do not have an associated workstation. If *WRKSTN is encountered in the resolution of output queue and printer device, *WRKSTN is replaced with *DEV if output queue is being resolved and *SYSVAL if printer device is being resolved.

Following are examples that illustrate where output goes from a job that runs in batch.

Example 1

Assume:

- There has not been a switch to an alternate user profile.
- The spooled file owner is *CURUSRPRF.
- The user's output queue (OUTQ) is OUTQ1 and the printer device (PRTDEV) is PRT1.
- The Submit Job (SBMJOB) command is used to submit the job to batch.
- The output queue parameter on the SBMJOB command is specified as *CURRENT.
- The printer device parameter on the SBMJOB command is specified as *CURRENT.

When the job runs in batch, the resulting spooled file is sent to OUTQ1 and that spooled file prints on the printer assigned to OUTQ1.

If no spooling was used when printing, the output would go to printer device PRT1.

The reason OUTQ1 and PRT1 are used is that *CURRENT is the value passed to the batch job by the user.

Example 2

Assume:

- There has not been a switch to an alternate user profile.
- The spooled file owner *CURUSRPRF value on the printer file has not been overridden.
- The user's output queue (OUTQ) is OUTQ1 and the printer device (PRTDEV) is PRT1.
- The Submit Job (SBMJOB) command is used to submit the job to batch.
- The output queue parameter on the SBMJOB command is specified as *USRPRF.
- The user profile has *WRKSTN as the value for the output queue parameter.
- The printer device parameter on the SBMJOB command is specified as PRT99.

When the job runs in batch, the resulting spooled file is sent to the output queue named PRT99 and the spooled file prints on PRT99. The *WRKSTN value for output queue is interpreted as *DEV and an output queue having the same name as the printer device is selected.

If no spooling was used when printing, the output would go to printer device PRT99.

Example 3

Assume:

- There has not been a switch to an alternate user profile.
- The spooled file owner *CURUSRPRF value on the printer file has not been overridden..
- The user's output queue (OUTQ) is OUTQ1 and the printer device (PRTDEV) is PRT1.
- The Submit Job (SBMJOB) command is used to submit the job to batch.
- The output queue parameter on the SBMJOB command is specified as *USRPRF.
- The user profile has *WRKSTN as the value for the output queue parameter.
- The printer device parameter on the SBMJOB command is specified as *WRKSTN.

When the job runs in batch, the resulting spooled file is sent to the system printer. This is because the *WRKSTN value for output queue is interpreted as *DEV and the printer device value of *WRKSTN is interpreted as *SYSVAL.

If no spooling was used when printing, the output would go to the printer defined as the system printer. This is the printer name assigned to the system value QPRTDEV.

Self-test: Determining output queue and printer device

Below are diagrams similar to the ones in the examples on the preceding pages. Read the information in the diagrams. Using the information you have acquired about the hierarchy of printing elements, determine what the output queue and printer device names would be.

Record the names of the output queue and the printer device in the empty blocks to the right of the diagram. The completed diagrams with the correct names for the output queues and printer devices are available on the next page.

Note: Keep in mind the SPOOL parameter value when deciding on your answer.

You should also assume the following for both self-tests:

- There has not been a switch to an alternate user profile.
- The spooled file owner attribute is *CURUSRPRF.

Job initiation

Job description

OUTQ(*USRPRF)

PRTDEV(PRT04)

User profile

OUTQ(PRT06)

PRTDEV(PRT01)

Workstation description

OUTQ(*DEV)

PRTDEV(*SYSVAL)

Job attributes

OUTQ (PRT06)

PRTDEV (PRT04)

Job runtime

Printer file

OUTQ(*JOB)

DEV(*JOB)

SPOOL=*YES

Job attributes

OUTQ(PRT06)

PRTDEV(PRT04)

Output queue

RBAFT506-0

Job initiation

Job description

OUTQ(PRT03)

PRTDEV(*USRPRF)

User profile

OUTQ(PRT02)

PRTDEV(*WRKSTN)

Workstation description

OUTQ(OUTQ1)

PRTDEV(PRT04)

Job attributes

OUTQ (PRT03)

PRTDEV (PRT04)

Printer file

OUTQ(PRT04)

DEV(*JOB)

SPOOL=*NO

Job attributes

OUTQ(PRT03)

PRTDEV(PRT04)

Printer device

RBAFT507-1

Self-test answers

Below are the diagrams from the preceding page, with the correct output queue and printer device parameter values filled in.

For the first diagram, the output queue name is PRT06.

Job initiation

Job description	User profile	Workstation description		Job attributes
OUTQ(*USRPRF) PRTDEV(PRT04)	OUTQ(PRT06) PRTDEV(PRT01)	OUTQ(*DEV) PRTDEV(*SYSVAL)	→	OUTQ (PRT06) PRTDEV (PRT04)

Job runtime

Printer file	Job attributes		Output queue
OUTQ(*JOB) DEV(*JOB) SPOOL=*YES	OUTQ(PRT06) PRTDEV(PRT04)	→	PRT06
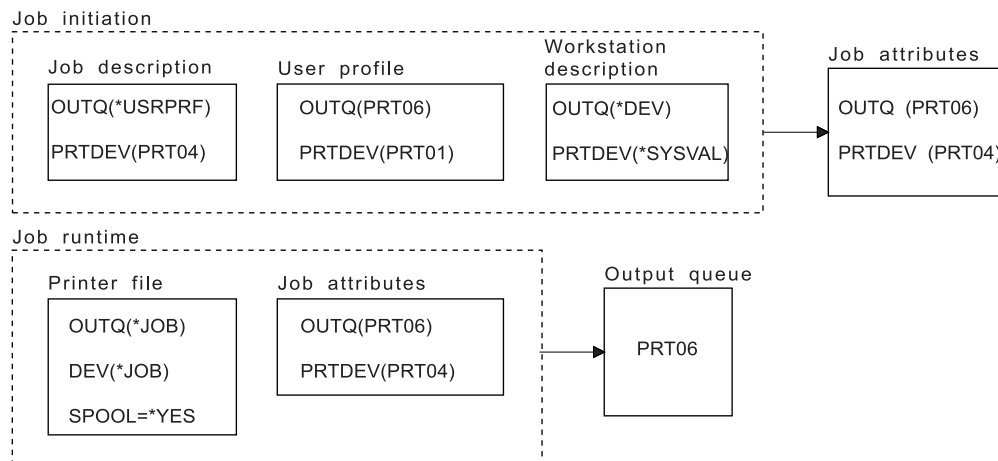
RBAFT508-0

The system looked first at the printer file and found SPOOL = *YES. Next it looked at the output queue value in the printer file, which is *JOB. It then looked at the output queue value in the job attribute OUTQ which is PRT06.

For the second diagram, the printer device value is PRT04.

Job initiation

Job description	User profile	Workstation description		Job attributes
OUTQ(PRT03) PRTDEV(*USRPRF)	OUTQ(PRT02) PRTDEV(*WRKSTN)	OUTQ(OUTQ1) PRTDEV(PRT04)	→	OUTQ (PRT03) PRTDEV (PRT04)

Job runtime — Printer file / Job attributes / Printer device

Printer file	Job attributes		Printer device
OUTQ(PRT04) DEV(*JOB) SPOOL=*NO	OUTQ(PRT03) PRTDEV(PRT04)	→	PRT04
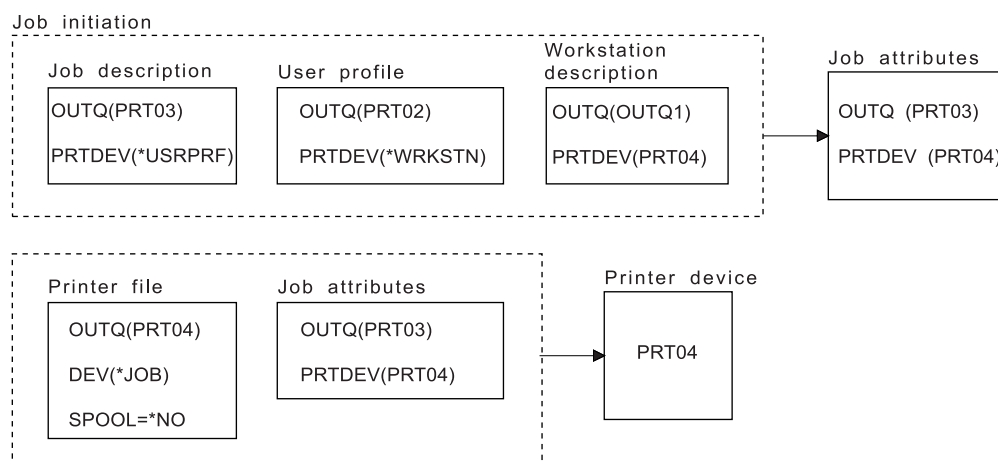
RBAFT509-0

Again, the system looked first at the printer file, but this time it found SPOOL = *NO. Next, it looked at the device value in the printer file, which is *JOB. It then looked at the device value in the job attribute PRTDEV.

The job attribute PRTDEV is PRT04.

If you had difficulty identifying the correct output queue and printer device you should go to "Controlling print activity" on page 13 and read the information again on controlling printing activity.

Where to find more examples

If you would like to experiment with some printing element examples, you can find them in Appendix A, "Examples of Working with Printing Elements".

- Creating output queues
- Displaying and changing your user profile
- Displaying system values (QPRTDEV)
- Changing a printer file
- Displaying a printer file
- Using the printer writer program
  - Starting a printer
  - Assigning a printer to a particular output queue
  - Stopping a printer
- Working with spooled files
  - Moving spooled files from one output queue to a different output queue
  - Holding spooled files

Part 2. Printer file and spooling support

Chapter 2. Printer file support

This chapter contains general-use programming interface and associated guidance information.

After reading this chapter you will:
- Understand what a printer file is.
- Understand how to create a printer file.
- Understand how to work with printer file overrides.
- Understand how to change printer files.
- Understand how a printer file interacts with an application program.

   An **application program** is used to perform a particular data processing task, such as inventory control, mailing labels, or payroll.

What is a printer file?

Printer files describe how the system is to operate on data as it passes between your application program and a printer. The printer file to be used is specified in your application program.

Different types of printer files

There are two different types of printer files:
- Program-described printer files

   Program-described printer files rely on the high-level language program to define records and fields to be printed.
- Externally-described printer files

   Externally-described printer files use data description specifications (DDS) rather than the high-level language to define records and fields to be printed. If you use DDS, you specify the name of your DDS source file in the SRCFILE parameter of the Create Printer File (CRTPRTF) command. DDS gives the application programmer much more format and print control over the printed output. For more information about DDS keywords, see DDS Reference in the iSeries Information Center.

What parameters make up a printer file?

Like other CL commands, the Create Printer File (CRTPRTF) command is made up of a group of parameters. These parameters, and the different values you can assign to each parameter, are designed to give you great flexibility in choosing how you want your printed output to look.

The following table provides a list of all the parameters of the CRTPRTF command. It also indicates if those parameters can be changed or temporarily overridden through the Override Printer File (OVRPRTF) command or through a program written in a high-level language (HLL) such as RPG, COBOL, PL/I, C, or BASIC.

Table 1. Printer Support Parameters

CL Parameter Name	Description	Specified on CRTPRTF and CHGPRTF Command	Specified on OVRPRTF Command	Specified in HLL Program
FILE	Printer file name	Qualified file name	*PRTF or file name	RPG/400®, COBOL/400®, PL/I, C/400*, and BASIC
TOFILE	Overriding file		*FILE or qualified name	
SRCFILE	Source file	*NONE or qualified name		
SRCMBR	Source member	*FILE or name		
GENLVL	Generation severity level	0, 10, 20, or 30		
OPTION	Type of output list	*SRC, *SOURCE, *NOSRC, *NOSOURCE, *LIST or *NOLIST, *SECLVL or *NOSECLVL		
DEV	Device name	*JOB, *SYSVAL, or device name	*JOB, *SYSVAL, or device name	
DEVTYPE	Device type	*SCS, *IPDS, *USERASCII, *AFPDS, *LINE, or *AFPDSLINE	*SCS, *IPDS, *USERASCII, *AFPDS, *LINE, or *AFPDSLINE	
CVTLINDTA	Convert line data and page definition to AFPDS	*NO, or *YES	*NO, or *YES	
PAGESIZE	Page length, width, and measurement method	Page length, width, and measurement method	Page length, width, and measurement method	RPG/400 and BASIC
LPI	Lines per inch	3, 4, 6, 7.5, 8, 9, or 12	3, 4, 6, 7.5, 8, 9, or 12	
CPI	Characters per inch	5, 10, 12, 13.3, 15, 16.7, 18 or 20	5, 10, 12, 13.3, 15, 16.7, 18 or 20	
FRONTMGN	Front margin	0 or any positive number	0 or any positive number	
BACKMGN	Back margin	*FRONTMGN, 0, or any positive number	*FRONTMGN, 0, or any positive number	
OVRFLW	Overflow line number	Line number	Line number	RPG/400 and BASIC
FOLD	Fold records	*NO, or *YES	*NO, or *YES	
RPLUNPRT	Replace unprintable characters	*YES and replacement character, or *NO	*YES and replacement character, or *NO	
ALIGN	Align forms	*NO, or *YES	*NO, or *YES	RPG/400
CTLCHAR	Control character	*NONE, *FCFC, or *MACHINE	*NONE, *FCFC, or *MACHINE	PL/I and C/400®
CHLVAL	Channel value	*NORMAL or channel value and line number	*NORMAL or channel value and line number	

Table 1. Printer Support Parameters (continued)

CL Parameter Name	Description	Specified on CRTPRTF and CHGPRTF Command	Specified on OVRPRTF Command	Specified in HLL Program
FIDELITY	Fidelity	*CONTENT or *ABSOLUTE	*CONTENT or *ABSOLUTE	
PRTQLTY	Print quality	*STD, *DRAFT, *NLQ, *DEVD, or *FASTDRAFT	*STD, *DRAFT, *NLQ, *DEVD, or *FASTDRAFT	
FORMFEED	Form feed mode	*DEVD, *CONT, *CONT2, *CUT, or *AUTOCUT	*DEVD, *CONT, *CONT2, *CUT, or *AUTOCUT	COBOL/400
DRAWER	Source drawer	1 through 255, *E1, or *FORMDF	1 through 255, *E1, or *FORMDF	
OUTBIN	Output bin	1 through 65,535	1 through 65,535	
FONT	Font identifier and point size	*CPI, *DEVD, font identifier name and point size of font if a point size is valid	*CPI, *DEVD, font identifier name and point size of font if a point size is valid	
CHRID	Character set code page	*DEVD, *CHRIDCTL, *SYSVAL, *JOBCCSID, or graphic character set and code page	*DEVD, *CHRIDCTL, *SYSVAL, *JOBCCSID, or graphic character set and code page	
DECFMT	Decimal Format	*JOB, *FILE	*JOB, *FILE	
FNTCHRSET	Font character set and point size	*FONT, font character set and code page, and point size of font if a point size is valid	*FONT, font character set and code page, and point size of font if a point size is valid.	
CDEFNT	Coded font and point size	*FNTCHRSET, coded font name, and point size	*FNTCHRSET, coded font name, and point size of font if a point size is valid.	
PAGDFN	Page definition	*NONE or page definition	*NONE or page definition	
FORMDF	Form definition	*NONE, *DEVD, or form definition	*NONE, *DEVD, or form definition	
AFPCHARS	AFP characters	*NONE or AFP characters	*NONE or AFP characters	
TBLREFCHR	AFP characters	*NO or *YES	*NO or *YES	
PAGRTT	Page rotate	*AUTO, *DEVD, *COR, 0, 90, 180, or 270	*AUTO, *DEVD, *COR, 0, 90, 180, or 270	
MULTIUP	Pages per side	1 through 4	1 through 4	
REDUCE	Reduce	*TEXT or *NONE	*TEXT or *NONE	
PRTTXT	Print text	*JOB, *SYSVAL, *BLANK, or print text	*JOB, *SYSVAL, *BLANK, or print text	
JUSTIFY	Justify	0, 50, or 100	0, 50, or 100	

Table 1. Printer Support Parameters (continued)

CL Parameter Name	Description	Specified on CRTPRTF and CHGPRTF Command	Specified on OVRPRTF Command	Specified in HLL Program
DUPLEX	Output printed on one or both sides of the paper	*NO, *YES, *TUMBLE, or *FORMDF	*NO, *YES, *TUMBLE, or *FORMDF	
UOM	Unit of measure	*INCH or *CM	*INCH or *CM	
FRONTOVL	Front side overlay	*NONE, qualified overlay name and library, offsets	*NONE, qualified overlay name and library, offsets	
BACKOVL	Back side overlay	*FRONTOVL, *NONE, qualified overlay name and library, offsets, constant back	*FRONTOVL, *NONE, qualified overlay name and library, offsets, constant back	
IPDSPASTHR	IPDS pass-through done for spooled file	*DEVD, *NO, or *YES	*DEVD, *NO, or *YES	
USRRSCLIBL	User resource libraries	*DEVD, *NONE, *JOBLIBL, or *CURLIB	*DEVD, *NONE, *JOBLIBL, or *CURLIB	
CORNERSTPL	Reference corner for corner staple	*NONE, *DEVD, *BOTRIGHT, *TOPRIGHT, *TOPLEFT, or *BOTLEFT	*NONE, *DEVD, *BOTRIGHT, *TOPRIGHT, *TOPLEFT, or *BOTLEFT	
EDGESTITCH	Specifies where staples will be placed	Reference Edge, Reference Edge Offset, Number of Staples, Staple Offsets	Reference Edge, Reference Edge Offset, Number of Staples, Staple Offsets	
SADLSTITCH	Specifies where staples will be placed	Reference Edge, Number of Staples, Staple Offsets	Reference Edge, Number of Staples, Staple Offsets	
FNTRSL	Font resolution	*DEVD, *SEARCH, 240, or 300	*DEVD, *SEARCH, 240, or 300	
DFRWRT	Defer write	*YES or *NO	*YES or *NO	
SPOOL	Spool data	*YES, or *NO	*YES, or *NO	
OUTQ	Output queue	*JOB, *DEV, or qualified name	*JOB, *DEV, or qualified name	
FORMTYPE	Form type	*STD or form type	*STD or form type	RPG/400
COPIES	Number of copies	Number of copies	Number of copies	
PAGERANGE	Starting and ending pages	Starting and ending pages	Starting and ending pages	
MAXRCDS	Maximum records	*NOMAX, or maximum records	*NOMAX, or maximum records	
FILESEP	Separator pages	Number of file separators	Number of file separators	
SCHEDULE	Schedule	*FILEEND, *JOBEND, or *IMMED	*FILEEND, *JOBEND, or *IMMED	
HOLD	Hold	*NO or *YES	*NO or *YES	

Table 1. Printer Support Parameters (continued)

CL Parameter Name	Description	Specified on CRTPRTF and CHGPRTF Command	Specified on OVRPRTF Command	Specified in HLL Program
SAVE	Save	*NO or *YES	*NO or *YES	
OUTPTY	Output priority	*JOB or output priority	*JOB or output priority	
USRDTA	User data	*SOURCE or user data	*SOURCE or user data	
SPLFOWN	Spooled file owner	*CURUSRPRF, *JOB, *CURGRPPRF, or *JOBGRPPRF	*CURUSRPRF, *JOB, *CURGRPPRF, or *JOBGRPPRF	
USRDFNOPT	User defined option	*NONE or user defined option	*NONE or user defined option	
USRDFNDTA	User defined data	*NONE or user defined data	*NONE or user defined data	
USRDFNOBJ	User defined object/type	*NONE or user defined object/ *DTAARA, *DTAQ, *FILE, *PSFCFG, *USRQ, *USRIDX or *USRSPC	*NONE or user defined object/ *DTAARA, *DTAQ, *FILE, *PSFCFG, *USRQ, *USRIDX or *USRSPC	
SPLFNAME	Spool file name		*FILE	
IGCDTA	Double-byte data	*NO or *YES	*NO or *YES	
IGCEXNCHR	Double-byte extension characters	*YES, or *NO	*YES, or *NO	
IGCCHRRTT	Rotate double-byte characters	*NO, or *YES	*NO, or *YES	
IGCCPI	Double-byte characters per inch	*CPI, 5, 6, 10, or *CONDENSED	*CPI, 5, 6, 10, or *CONDENSED	
IGCSOSI	Shift control characters	*YES, *NO, or *RIGHT	*YES, *NO, or *RIGHT	
IGCCDEFNT	Double-byte coded font and point size	*SYSVAL, coded font name, and point size	*SYSVAL, coded font name, and point size	
WAITFILE	File wait time	*IMMED, *CLS or number of seconds	*IMMED, *CLS or number of seconds	RPG/400 and PL/I
SECURE	Secure from other overrides		*YES or *NO	
SHARE	Shared file	*NO or *YES	*NO or *YES	PL/I
LVLCHK	Level check	*YES or *NO	*YES or *NO	RPG/400, COBOL/400, PL/I, C/400, and BASIC
AUT	Authority	*CHANGE, *ALL, *USE, *EXCLUDE or authorization list name		
REPLACE	Replace	*YES or *NO		
TEXT	Text	*SRCMBRTXT, *BLANK, or text description		

CL Parameter Name	Description	Specified on CRTPRTF and CHGPRTF Command	Specified on OVRPRTF Command	Specified in HLL Program
	Record length			RPG/400, COBOL/400, PL/I, C/400, and BASIC
	End-of-page indicator			RPG/400, PL/I and BASIC
	Skip before	Line number		RPG/400
	Skip after	Line number		RPG/400
	Space before	Number of lines		RPG/400, COBOL/400
	Space after	Number of lines		RPG/400, COBOL/400

Creating a printer file

Printer files can be created by the user or the user could choose to use the printer files supplied by IBM.

IBM-supplied printer files

The following list contains the IBM-supplied printer files:

- QSYSPRT: a program-described printer file in library QSYS.
- QPRINT: Default spooled output printer file for normal print.
- QPRINTS: Default spooled output printer file for special forms.
- QPRINT2: Default spooled output printer file for two copy output.
- QPSPLPRT: Default spooled output printer file for the spooling subsystem.

User-created printer files

You use the Create Printer File (CRTPRTF) command to begin the process of creating a printer file tailored specifically to your needs.

If, at this time, you want to become familiar with this command, type CRTPRTF and press F4 (Prompt). Or you can go to "What parameters make up a printer file?" on page 41 and view a chart that lists all the printer file parameters.

The next section of this chapter contains examples where an application program calls (uses) a user-created printer file. The examples focus on a program-described printer file and an externally described printer file.

Why is the concept of externally described or program-described printer files important?

The reason the terms *externally described* or *program-described* are important is that by specifying *NONE on the SRCFILE parameter of the CRTPRTF command, the user is telling the system that the high-level language the application program is written in will control how the printed output appears on a printed page.

If the user specifies a source file name and member (made up of data description specifications) on the SRCFILE parameter, then the application program will use information that is specified in the member to control how the printed output appears on a printed page.

To learn how printer files interact with application programs, see the following examples:

- Program described: "Using a program-described printer file with an application program"
- Externally described: "Using an externally-described printer file with an application program" on page 59

Using a program-described printer file with an application program

This example shows how an application program calls and uses a program-described printer file to control how your printed output will look.

This example consists of:

- An RPG-coded application program that produces mailing labels.
- A printer file that is opened by the application program when the application program runs.
- A detailed listing of which parameters from the printer file are used by the application program and at what point during the processing of the application program they are used.

This is the RPG-coded application program. The reverse-image numbers within the program correspond to the list on the following pages. That list explains how this program works and specifically how the program opens and uses the printer file.

- Part **1** opens the printer file
- Part **2** sends the output to the designated output queue
- Part **3** closes the opened files
- Part **4** processes the data
- Part **5** provides the code to control how the printed output will look

```
*********************************************************************
          FADDRESS IF  E          K        DISK
1    FLABELPRTO  F   132    OF    PRINTER
*********************************************************************

                CTAR       20 1
                CSAR       30 1
                STAR        2 1
                ZPAR        8 1

                READ ADDRESS                     10

        *IN10   DOWEQ'0'

        ADD2    IFEQ *BLANKS
                MOVE '1'      *IN55
                ELSE
                MOVE '0'      *IN55
                END

                EXSR CKCITY

*********************************************************************
2               EXCPTPRINT
*********************************************************************
                READ ADDRESS          10
```

```
                            END

          ********************************************************************
3           MOVE '1'      *INLR
          ********************************************************************

                   CKCITY   BEGSR

                            MOVEA*BLANKS   CTAR
                            MOVEA*BLANKS   STAR
                            MOVEA*BLANKS   ZPAR
                            MOVEA*BLANKS   CSAR

                            MOVEACITY      CTAR
                            MOVEAST        STAR
                            MOVEAZIP       ZPAR

                            Z-ADD1         X      20
                            Z-ADD1         Y      20

                            EXSR LOOKBL                    1ST WORD.
                            ADD  1         X
                   CTAR,X   IFGT *BLANKS                   2ND WORD
                            MOVE ' '       CSAR,Y
                            ADD  1         Y
                            EXSR LOOKBL
4                  ADD  1        X
                   CTAR,X   IFGT *BLANKS                        3RD WORD
                            MOVE ' '       CSAR,Y
                            ADD  1         Y                    EXSR LOOKBL
                            END
                            END
                            MOVE ','       CSAR,Y
                            ADD  1         Y
                            MOVE ' '       CSAR,Y
                            ADD  1         Y
                            MOVE STAR,1    CSAR,Y
                            ADD  1         Y

                            MOVE STAR,2    CSAR,Y
                            ADD  1         Y
                            MOVE ' '       CSAR,Y
                            ADD  1         Y
                            MOVE ' '       CSAR,Y
                            ADD  1         Y
                            Z-ADD1         X
                   X        DOWLT9
                            MOVE ZPAR,X    CSAR,Y
                            ADD  1         Y
                            ADD  1         X
                            END
                            MOVEACSAR      CTSTZP 30
                            ENDSR

                            LOOKBL         BEGSR
                   CTAR,X         DOWGT*BLANKS
                            MOVE CTAR,X    CSAR,Y
                            ADD  1         X
                            ADD  1         Y
                            END
                            ENDSR
          ********************************************************************
                            RTE    1 2        PRINT
                            NAME   25
                            E  1              PRINT
                            ADD1   25
5                  E  1  N55        PRINT
                            ADD2      25
```

```
                        E  1                PRINT
                        CTSTZP 30
                        E  1   55           PRINT
        ****************************************************************
```

Open processing

Part **1** of the application program opens files that are called by the application program.

When a program opens files, an object of type *FILE is connected to the program for processing. Among the files opened in this example, and of particular interest at this point, is the printer file whose name is LABELPRT. You can find the printer file name LABELPRT next to the **1** in the program listing.

A printer file is opened to prepare the system so that the application can put data into a spooled file or print it out directly to a printer. Information from the high-level language application program, the printer file, and any printer file overrides is combined.

The printer file open operation is controlled by parameters specified in the printer file, the high-level language program, and in printer file overrides (through the OVRPRTF command). See "Overriding printer files" on page 69 for more information on overrides.

As an example, if the printer file specified lines per inch (LPI) of 8, and an OVRPRTF command specified an LPI of 6, the LPI of 6 would be used since the override value specified by the OVRPRTF command takes precedence over the LPI value specified in the printer file.

Printer file parameters

The following list contains parameters from the printer file LABELPRT. These are the parameters that the application program accesses or looks at when it opens the printer file. They are the majority of the parameters in the printer file, but not all of them. When the application looks at each parameter, it finds a value specified for each parameter. Unless otherwise noted, the parameters also apply to externally-described printer files.

Some of the values are discussed in more detail than others, because some of the parameters have many more possible values than others.

FILE
> Specifies the qualified name by which the printer file being called is known.

DEV
> Specifies the name of a printer device description. The device description is an object that contains information describing a particular device (printer in this case) or logical unit that is attached to the system. For nonspooled output, this identifies the printer device used with the printer file to produce the printed output. For spooled output, if OUTQ(*DEV) is also specified, the default output queue for the specified printer is used for the spooled file.

DEVTYPE
> Specifies the type of data stream created for a printer file. This parameter indicates whether the resulting data stream should be:
> * Intelligent printer data stream (*IPDS)
> * SNA character stream (*SCS),
> * ASCII data stream (*USERASCII)

- Advanced Function Printing data stream (*AFPDS)
- Line data (*LINE)
- Mixed data (*AFPDSLINE)

For more information on the DEVTYPE parameter, go to "Using the device type (DEVTYPE) parameter" on page 75.

CVTLINDTA

Specifies whether line data and a page definition should be converted to AFPDS as the data is written to spool. This parameter is intended for use by those applications (RPG, COBOL, etc.) that are currently producing line data. It can also be used with new Java applications that use the new Java line data print classes.

PAGESIZE

Specifies the length and width of the printer forms used by this printer file.

Go to Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377 to view tables that show:

- Range of values for lines per page for each printer type and for each value of lines per inch (LPI) valid for the printer.
- Range of values for characters per line for each printer type and for each value of characters per inch (CPI) for the printer.

Notes:

1. For IPDS printers configured with AFP(*NO) on the Create Device Description (Printer) (CRTDEVPRT) or Change Device Description (Printer) (CHGDEVPRT) CL commands, the printer file parameters are used to override the forms size set on the printer. For printers configured with AFP(*YES), the forms size set on the printer is used to determine the actual size of the forms loaded in the printer.

2. If *ROWCOL is specified as the method of measure on the PAGESIZE parameter, and either a coded font or font character set is specified on the printer file, the page width is calculated using 10 characters per inch.

3. If the PAGESIZE parameter value is *UOM, the page size length and width is converted to *ROWCOL (number of rows and columns per page) when spooled files are created.

   If the DEVTYPE is *IPDS, the pitch of the font specified by the FONT parameter is used to calculate the number of columns allowed per page. For example, assume:

   > Form size is 8.5 x 11 inches
   >
   > The UOM parameter value is *INCH
   >
   > The CPI parameter value is 10 (the default)
   >
   > The FONT parameter value is 86

   The number of columns available for printing is 102 (8.5 inches x 12 characters per inch), because FONT 86 is a 12-pitch font. The CPI parameter is ignored.

   If the DEVTYPE is *SCS, the CPI parameter is used for the conversion. The CPI value should match the pitch of the specified FONT, or the FONT parameter should be *CPI. Let's use the same values as in the previous example but have the DEVTYPE be *SCS and the FONT value be *CPI. Now, because the CPI parameter is 10, the printed output will be truncated at 85 columns (8.5 inches x 10 characters per inch).

LPI

Specifies the line spacing setting on the printer, in lines per inch, used by this printer file.

UOM

Specifies the unit of measure to be used for various measurements used in printing. Possible values are:

- *INCH (Use inches as the unit of measure.)
- *CM (Use centimeters as the unit of measure.)

CPI

Specifies the printer character density, in characters per inch (CPI), used by this printer file.

For more information on CPI, go to "Printer font support" on page 103.

OVRFLW

Specifies the line number where an overflow message is issued.

RPLUNPRT

Specifies (1) whether unprintable characters are replaced and (2) which substitution character (if any) is used. An unprintable character is a character the printer is unable to print. For more information on replacing unprintable characters, go to "Replacing on unprintable characters" on page 106.

FIDELITY

Specifies whether the spooled file should continue printing if errors are encountered.

*CONTENT

If errors are encountered, the spooled file continues to print.

*ABSOLUTE

If errors are encountered, the spooled file does not print.

For more information on FIDELITY, go to "Using the fidelity (FIDELITY) parameter" on page 88.

CTLCHAR

Specifies whether the printer file supports input with print control characters. Incorrect control characters encountered are ignored, and single spacing is assumed. Control characters (such as carriage return, font change, or end of transmission) are contained in the data stream that is being prepared for printing.

PRTQLTY

Specifies, for the 4214, 4224, 4230, 4234, 4247 and 5219 Printers, the quality of print produced.

FORMFEED

Specifies, for the 3912, 3916, 4028, 4214, 4224, 4230, 4247 5219, and 5553 Printers, the form feed attachment used by this printer file.

DRAWER

Specifies the source drawer used when automatic cut sheet feed mode is used (specified by FORMFEED(*AUTOCUT)). When you specify a form definition, the value specified for the Drawer parameter will override the drawer value specified in the form definition. If you want to use the drawer value specified for the Form definition parameter, you must specify *FORMDF for the Drawer parameter.

OUTBIN

Specifies the destination bin of the output on printers that support multiple output bins. Possible values are:

*DEVD

The destination of the output is the default output bin for the printer device.

Output bin

For printed output, specifies the output bin to be used on the printer device.

Valid values range from 1 through 65535.

FONT

Specifies the font identifier and the point size (if the font supports multiple point sizes) of the font that is used for the following printers:

- 3130, 3812, 3816, 3820, 3825, 3827, 3829, 3831, 3835, 3900, 3912, 3916, 3930, 3935, 4028, 4224, 4230, 4234, 4247, 4312, 4317, 4324, 5219
- InfoPrint 20, InfoPrint 32, InfoPrint 3000, and InfoPrint 4000 printers. This includes ASCII printers that emulate the 3812 or 5219 Printer.

Notes:

1. For a list of fonts that are supported, see Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377.

2. For more information about fonts and how various printers use them, see "Printer font support" on page 103

CHRID

Specifies the character identifier (graphic character set and code page) for fields identified with the CHRID DDS keyword.

Notes:

1. If the CHRID parameter value is *JOBCCSID, the character set and code page are determined by the CCSID value of that job.

   *JOBCCSID indicates that all data being printed should be printed using the CCSID value for the current job. With externally-described printer files, constants specified in DDS are translated from the CCSID of the DDS source file to the CCSID of the current job. In addition, the CHRID DDS keyword is ignored.

2. When you specify *CHRIDCTL, the printer checks the CHRIDCTL job definition attribute to determine whether to use *JOBCCSID or *DEVD on the current job.

3. For a list of the character IDs that are supported, see Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377.

4. For more information on multinational environments and alternative character sets and code pages, go to "Using alternative character sets and code pages for printer output" on page 107.

DECFMT

Specifies which decimal format value is used when editing numeric fields in the printer file. Possible values are:

*JOB Use the decimal format from the DECFMT job attribute at the time the file is opened.

*FILE   Use the decimal format that is stored with the file when the file was created.

FNTCHRSET

Specifies the font character set, code page, and point size to be used at printing time.

For raster fonts, the font character set identifier provides the size and style of the character set. For outline fonts, the font character set identifier provides the style of the character set. A point size is required to specify the size of the outline font.

The code page identifies which characters in a font are available for printing. For more information about font character sets and code pages, go to "Using the font character set (FNTCHRSET) parameter" on page 99.

Note: Do not specify an AFP Unicode migration font character set and code page for this printer file parameter. Specify a Unicode font using the FNTCHRSET DDS keyword in the DDS Reference: Printer Files in the iSeries Information Center.

CDEFNT

Specifies the coded font and point size to be used at printing time.

A coded font is an AFP resource that is made up of a font character set name and a code page name. A point size is required for outline fonts.

For more information about coded fonts and their relationship to font character sets, code pages, and font global identifiers (FGIDs), go to "Using the coded font (CDEFNT) parameter" on page 100.

PAGDFN

Specifies the page definition to be used to format line data.

Print Services Facility/400 (PSF/400) requires a page definition when *LINE or *AFPDSLINE is specified for the device type. An inline page definition gets built from the printer file parameters if you specify either *LINE or *AFPDSLINE, and you specify *NONE for the page definition.

A page definition is also required for *USERASCII spooled files that contain XML.

FORMDF

Specifies the form definition to use when printing a file. A form definition is a resource object that defines the characteristics of the form. Characteristics include duplexing, overlays, position of page data on the form, number of copies, modification of pages, and input drawer. When printing to a printer that is configured as AFP(*YES), a form definition is required. If you do not provide a form definition, an inline form definition will be built from the printer file parameters.

AFPCHARS

Specifies one or more AFP characters (coded fonts) to use with line data and a page definition.

TBLREFCHR

Specifies whether or not table reference characters are present in the line data.

When you use forms control characters with the data, the table reference character follows the forms control characters, but precedes the data bytes. If you do not specify forms control characters, the table reference character is the

first byte of the data record. As with forms control characters, if you use table reference characters, every data record must contain a table reference character byte.

PAGRTT

Specifies the degree of text rotation for the following printers: 3160, 3130, 3812, 3816, 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900, 3916, 3930, 3935, 4028, 4312, 4317, 4324. It also specifies the degree of text rotation for the InfoPrint 20, InfoPrint 32, InfoPrint 3000, and the InfoPrint 4000. This parameter allows the user to specify the degree of text rotation on the page with respect to how you load the form into the printer.

Note: Overlays and page segments do not rotate with the rest of the page.

For more information and examples on overlays and rotation, go to "Using overlays and rotation" on page 90.

PRTTXT

Specifies the printing of a line of text at the bottom of each page. For more information about printing information on the bottom of your printed page, go to "Print text" on page 109.

JUSTIFY

For the 5219 and 3812 SCS only, specifies the printing positions of the characters on a page so the right-hand margin of printing is regular.

DUPLEX

Specifies whether the output is printed on only one or on two sides of the paper. When you specify a Form definition, the value specified on the Duplex parameter will override the duplex value specified in the form definition. If you want to use the duplex value specified in the form definition, you must specify *FORMDF for the duplex parameter.

IPDSPASTHR

Specifies whether IPDS pass-through occurs for the spooled file. Not all SCS or IPDS spooled files are eligible for IPDS pass-through. They may contain special functions that require transformation to AFPDS to print correctly. The PSF configuration object's IPDSPASTHR parameter will be used when you specify *DEVD. The PSF configuration object value will be ignored for any other specified value.

USRRSCLIBL

Specifies the list of user resource libraries to use for searching for AFP resources for a spooled file. You can specify either a maximum of four library names, which will replace the spooling job's library list, or you can specify one of the following values:

- *CURLIB
- *DEVD
- *JOBLIBL
- *NONE

CORNERSTPL

Specifies the reference corner that is used for a corner staple. Refer to your printer's documentation to find out which values your printer supports.

EDGESTITCH

Specifies the placement of the staples along the finishing margin. Refer to your printer's documentation to find supported values for this parameter. For more information, see "Using the edge stitch (EDGESTITCH) parameter" on page 96.

SADLSTITCH

Specifies where to place staples along the finishing margin. The position of the finishing margin for this parameter is along the center of the media and parallel to the reference edge. Refer to your printer's documentation to find supported values for this parameter. For more information, see "Using the saddle stitch (SADLSTITCH) parameter" on page 98.

FNTRSL

If the resolution is absent from the data stream, this parameter specifies the resolution that PSF/400 uses when printing to a multiple resolution printer. If *DEVD is specified, the value from the PSF configuration object's FNTRSL parameter is used. Any other value causes the PSF configuration object value to be ignored. For more information about searching a library list for a font resource, see "User and device resource library lists" on page 246

SPOOL

Specifies whether the output data for the printer file is spooled. If SPOOL(*NO) is specified, the following parameters in the printer file command are ignored: OUTQ, COPIES, MAXRCDS, FILESEP, SCHEDULE, HOLD, SAVE, OUTPTY, and USRDTA.

SCHEDULE

Specifies, for spooled files only, when the spooled file is made available to the printer writer program.

For more information on how the SCHEDULE parameter is used to control when the spooled file is made available to a printer writer to produce a printed document, go to "Using the SCHEDULE parameter" on page 76.

USRDTA

Specifies, only if the file is spooled, some user-specified data that identifies the file on the output queue.

SPLFOWN

Specifies the owner of the spooled file. You use this parameter only for spooled output.

USRDFNOPT

Specifies one or more user-defined options to be used by user applications or user-specified programs that process spooled files.

USRDFNDTA

Specifies user-defined data to be used by user applications or user specified programs that process spooled files.

USRDFNOBJ

Specifies a user-defined object and type to be used by user applications or user specified programs that process spooled files. The object type can be a data area, data queue, file, user index, PSF configuration object, user queue, or user space.

IGCDTA

Specifies, for program-described printer files, whether the printer file processes double-byte character set data. For externally described printer files, specifies double-byte character set or UCS-2 attributes of the printer file.

IGCEXNCR

Specifies whether the system processes double-byte character extension characters.

IGCCHRTT
Specifies whether the printer should rotate double-byte characters 90 degrees counterclockwise when printing.

IGCCPI
Specifies the printer character density of double-byte characters, in characters per inch.

IGCSOSI
Specifies how the system prints shift control characters.

IGCCDEFNT
Specifies the font and point size that the system uses to print DBCS data to an IPDS printer. The point size is for outline fonts only. The printer must be configured AFP(*YES). For more information on DBCS coded fonts, go to "Using the DBCS coded font (IGCCDEFNT) parameter" on page 99.

WAITFILE
Specifies the number of seconds the program waits for allocation of the file resources when the file is opened.

SHARE
Specifies whether the open data path (ODP) for the printer file can be shared with other programs in the same routing step.

LVLCHK
Specifies whether the level identifiers of the record formats in this printer file are checked when the file is opened by the program. For more information on how this parameter works to redirect files, go to "Effect of changing fields in a file description" on page 111.

AUT
Specifies the authority granted to users who do not have specific authority to the file, who are not on the authorization list, and whose users' group has no specific authority to the file.

TEXT
Specifies text that briefly describes the printer file.

Output processing

Part **2** of the application program performs the operations of reading, compiling, and sending the output to the output queue specified in the OUTQ parameter of the CRTPRTF command or to the printer specified in the DEV parameter of the CRTPRTF command. In this example, the SPOOL parameter has a value of (*YES). This means the output will become a spooled file in the designated output queue.

The following printer file parameters are the CRTPRTF parameters looked at by the system program and application program during the output processing portion of the application program. Unless otherwise noted, these parameters also apply to externally-described printer files.

SRCFILE
Specifies the qualified name of the source file and member, if one exists, that contains the data description specifications (DDS).

This example does <u>not</u> use a DDS source file. Thus, when the application program calls the printer file LABELPRT and looks at the SRCFILE parameter, the value will be *NONE.

Since no DDS will be used, the high-level language must provide the programming to control how the printed output will look. This example contains the high-level language code in part **5** of the example program listing.

SRCMBR
Specifies the name of the member in the source file that contains the DDS for this printer file.

FOLD
Specifies whether all positions in a record are printed when the record length exceeds the page width (specified by the PAGESIZE parameter).

ALIGN
Specifies whether the page alignment must be verified by the operator before printing is started. For more information about the align parameter and its relationship to the Start Printer Writer (STRPRTWTR) command, go to "Using the align (ALIGN) parameter" on page 77.

CHLVAL
Specifies a list of channel numbers with their assigned line numbers. Use this parameter only if CTLCHAR(*FCFC) has been specified. For more information on first-character forms-control data, go to "First-character forms-control data" on page 101.

PRTTXT
Specifies the printing of a line of text at the bottom of each page. For more information about printing information on the bottom of your printed page, go to "Print text" on page 109.

REDUCE
Specifies whether or not to reduce the output when doing multiple up printing. Multiple up printing refers to the Pages Per Side parameter of the printer file. Possible values are:

*TEXT
Reduce the text output when doing multiple up printing.

*NONE
Do not reduce the output when doing multiple up printing.

> **Note:** If a printer has a hardware feature that supports multiple up printing, *NONE allows the use of that hardware feature.

MULTIUP
Specifies how many logical pages of output can be printed on one physical page of paper. The values are 1 through 4. 1 is the default value.

For example: if you select 4 as the parameter value and you are printing on both sides of the paper, you actually print 8 pages of output on 1 piece of paper.

For more information on the MULTIUP parameter, go to "Using the MULTIUP(1, 2, 3, or 4) and REDUCE(*TEXT) parameters" on page 81.

FRONTMGN
Specifies the margin to be used on the front side of a piece of paper.

For more information on front margins, go to "Using the margin (FRONTMGN and BACKMGN) parameters" on page 93.

BACKMGN
Specifies the margin to be used on the back side of a piece of paper.

For more information on back margins, go to "Using the margin (FRONTMGN and BACKMGN) parameters" on page 93.

FRONTOVL
Specifies the overlay to be used on the front side of a piece of paper. Possible values are:

- *NONE (No overlay is used.)
- qualified overlay-name (The name of the overlay to be used and the library the overlay is in.)

Front side overlays are valid only for printers configured for Advanced Function Printing*. For more information on overlays, go to "Using the overlay (FRONTOVL and BACKOVL) parameters" on page 89.

BACKOVL
Specifies the overlay to be used on the back side of a piece of paper. Possible values are:

- *NONE (No overlay is used.)
- Qualified overlay-name (The name of the overlay to be used, the library the overlay is in, and whether to use the constant back function.)
- *FRONTOVL (Use the overlay specified for the FRONTOVL parameter.)

Back side overlays are valid only for printers configured for advanced function printing. For more information on overlays, go to "Using the overlay (FRONTOVL and BACKOVL) parameters" on page 89.

MAXRCDS
Specifies, for spooled output only, the maximum number of records in the spooled file for spooled jobs using this printer file.

DFRWRT
DFRWRT(*YES) means the system controls the amount of output held in the buffer before being sent to the printer.

For spooled files, DFRWRT(*NO) means spooled data is sent to the printer when a page of output is available or when the system buffer is full. (Buffers are usually 512 bytes.) DFRWRT(*NO) has effect on spooled files only if SCHEDULE(*IMMED) is specified.

For direct output, DFRWRT(*NO) means data is sent to the printer each time the application program performs a write operation.

Close processing

Part **3** of the application program performs the close operations of the application program.

When the application program has finished the output processing part of the application program, it performs a close operation on all the files it opened during the open processing portion of the application program.

The following printer file parameters from the CRTPRTF command are the parameters looked at by the system program and the application program during the close portion of the application program.

SCHEDULE
Specifies, for spooled files only, when the spooled file is made available to the printer writer program.

For more information on how the SCHEDULE parameter is used to control when the spooled file is made available to a printer writer to produce a printed document, go to "Using the SCHEDULE parameter" on page 76.

Using an externally-described printer file with an application program

This example shows how an application program calls and uses an externally described printer file to control how your printed output will look.

This example will consist of:
- An RPG-coded application program that produces mailing labels.
- A printer file that is opened by the application program when the application program runs.
- A detailed listing of which parameters from the printer file are used by the application program and at what point during the processing of the application program they are used.
- A detailed listing of the DDS keywords and an explanation of which DDS keywords are used by the mailing label application program.

This is the RPG-coded application program. The reverse-image numbers within the program correspond to the list on the following pages. That list explains how this program works and specifically how the program opens and uses the printer file.
- Part **1** opens the printer file
- Part **2** sends the output to the designated output queue
- Part **3** closes the opened files
- Part **4** processes the data

```
********************************************************************
                 FADDRESS IF  E          K       DISK
 1    FLABELPR30   E                      PRINTER
********************************************************************

                 CTAR       20 1
                 CSAR       30 1
                 STAR        2 1
                 ZPAR        8 1

                 READ ADDRESS                    10
********************************************************************
                 WRITEHEADNG

       *IN10     DOWEQ'0'

                 EXSR CKCITY

 2               WRITEDETAIL1

       ADD2      IFNE *BLANKS
                 WRITEDETAIL3
                 END

                 WRITEDETAIL4
                 READ ADDRESS                    10

                 END
********************************************************************

********************************************************************
 3               MOVE '1'      *INLR
********************************************************************
        CKCITY   BEGSR
```

```
              MOVEA*BLANKS   CTAR
              MOVEA*BLANKS   STAR
              MOVEA*BLANKS   ZPAR
              MOVEA*BLANKS   CSAR
              MOVEACITY      CTAR
              MOVEAST        STAR
              MOVEAZIP       ZPAR
              Z-ADD1         X        20
              Z-ADD1         Y        20

              EXSR LOOKBL                      1ST WORD

              ADD  1         X
      CTAR,X  IFGT *BLANKS                     2ND WORD
              MOVE ' '       CSAR,Y
              ADD  1         Y
              EXSR LOOKBL
              ADD  1         X
      CTAR,X  IFGT *BLANKS                     3RD WORD
              MOVE ' '       CSAR,Y
              ADD  1         Y
  4           EXSR LOOKBL
              END
              END
              MOVE ','       CSAR,Y
              ADD  1         Y
              MOVE ' '       CSAR,Y
              ADD  1         Y
              MOVE STAR,1    CSAR,Y
              ADD  1         Y
              MOVE STAR,2    CSAR,Y
              ADD  1         Y
              MOVE ' '       CSAR,Y
              ADD  1         Y
              MOVE ' '       CSAR,Y
              ADD  1         Y
              Z-ADD1         X
      X       DOWLT9
              MOVE ZPAR,X    CSAR,Y
              ADD  1         Y
              ADD  1         X
              END
              MOVEACSAR      CTSTZP 30
              MOVEACSAR      CTSTZ2 30
              ENDSR

      LOOKBL  BEGSR
      CTAR,X  DOWGT*BLANKS
              MOVE CTAR,X    CSAR,Y
              ADD  1         X
              ADD  1         Y
              END
              ENDSR


  ********************************************************************
```

Open processing

Part **1** of the application program opens files that are called by the application
program. Among those, and of particular interest at this point, is the printer file
whose name is LABELPR3. You can locate LABELPR3 next to **1** in the program
listing.

A printer file is opened to prepare the system so that the application can put data
into a spooled file or print it out directly to a printer. Information from the
high-level language application program, the printer file, and any printer file
overrides is combined.

The printer file open operation is controlled by parameters specified in the printer file, the high-level language program, and in printer file overrides (through the OVRPRTF command). See "Overriding printer files" on page 69 for more information on overrides.

As an example, if the printer file specified lines per inch (LPI) of 8, and an OVRPRTF command specified an LPI of 6, the LPI of 6 would be used since the override value specified by the OVRPRTF command takes precedence over the LPI value specified in the printer file.

Output processing

Part **2** of the application program performs the operations of reading, compiling, and sending the output to the output queue specified in the OUTQ parameter of the CRTPRTF command, or to the printer specified in the DEV parameter of the CRTPRTF command. In this example, the SPOOL parameter has a value of (*YES) which means the output will become a spooled file in the designated output queue.

Unless otherwise noted, the printer file parameters listed in Output processing on page 56 are also valid for externally-described printer files.The following printer file parameters from the CRTPRTF command are additional parameters that are looked at by the application program during the output processing.

SRCFILE
Specifies the qualified name of the source file and member, if one exists, that contains the data description specifications (DDS).

Since this example uses DDS, look at **1** in the program listing and see that the name of the printer file is LABELPR3. LABELPR3 was compiled using the source from the member and file that are listed here. See "DDS keywords" on page 62 for an example of the compiled DDS and a list of DDS keywords.

Note: The DDS will be compiled before the application program runs. The application program never looks at the DDS file and member, only at the compiled results.

Option
Specifies the type of printout that will be produced when the printer file is created.

GENLVL
Specifies the severity level of DDS messages that cause file creation to fail.

Data description specifications
Below is the example of the compiled DDS used by the RPG program. You can update the DDS; however, you must then re-compile it.

```
000100900115                 R HEADNG
000200900115                                    3  2'MAILING LABELS'
000300900115
000400900115                 R DETAIL1
000500900115                   NAME      25     2  2UNDERLINE
000600900115                   ADD1      25     3  2
000700900115                 R DETAIL3
000800900115                   ADD2      25        2SPACEB(1)
000900900115                 R DETAIL4
001000900115                   CTSTZP    30        2HIGHLIGHT SPACEB(1)
```

This example uses three DDS keywords: SPACEB, UNDERLINE, and HIGHLIGHT.

DDS keywords: Below is a listing of all available DDS keywords associated with printer files.

Remember: DDS and its associated keywords can only be used if the SRCFILE parameter contains the name of the file and the SRCMBR parameter contains the name of the member that the DDS source resides in.

The following list contains all the DDS keywords that you can use to help you control how your printed output will look. Remember, some DDS keywords can only be used if the target printer supports them. For example: COLOR can only be specified if the target printer is a 4224.

For more detailed information on how to create and code DDS source files and to find out which printers support which DDS keywords, see the DDS Reference: Printer files topic in the iSeries Information Center.

ALIAS
Alternative Name. Use this field-level keyword to specify an alternative name for a field. When the program is compiled, the alternative name is brought into the program instead of the DDS field name. The high-level language compiler in use determines if the alternative name is used. Refer to the appropriate high-level language reference manual for information about ALIAS support for that language.

BARCODE
Bar Code. Use this field-level keyword to print a field as a user-specified bar code. BARCODE is valid for IPDS printers. (A **bar code** is a pattern of bars of various widths containing data to be interpreted by a scanning device.)

BLKFOLD
Blank Fold. Use this field-level keyword for named fields that overflow onto subsequent print lines, to cause folding to occur at a blank rather than at the end of a line. If the blank fold keyword is not specified, the line folds at the end of the physical print line.

BOX
Box. Use this record-level keyword to specify the position on a page where a box should be drawn. You can specify the corners and the weight (thickness) of the lines in the box. Additionally, you can specify the color of the lines of a box and whether to have box shading. If you select box shading, you can also specify the color of the box shading.

CCSID
Coded Character Set Identifier. Use this file-, record-, or field-level keyword to specify that a 'G' type field supports UCS-2 data instead of DBCS graphical data.

CDEFNT
Coded Font. Use this record- or field-level keyword to specify the coded font and point size for printing a named or constant field or fields within a record.

CHRID
Character identifier. Use this field-level keyword to specify that a graphic character set and code page other than the device default can be used for this field. This can be important when extended alphabetics (characters such as u with an umlaut or c with a cedilla) are to be printed.

CHRSIZ
Character Size. Use this record- or field-level keyword to expand the width

and height of a record or field. This applies to SCS double-byte characters, and IPDS and AFPDS single-byte characters.

COLOR

Color. If the printer device supports it, use this field-level keyword, to specify the color for a field. If you do not specify COLOR, or if the keyword is not valid for a printer device, black (the default value) is used.

CPI

Characters per Inch. This record- or field-level keyword specifies the horizontal printing density for the record format or field you are defining. Use CPI to:

- Darken logos and other printed graphics that you create using the DFNCHR keyword.
- Place more data in less space on printed forms.
- Fit the appearance of a form to your needs.

CVTDTA

Convert Data. This field-level keyword converts character data to hexadecimal data when the field is passed to the printer. You can use the CVTDTA keyword to define:

- Logos or emblems for a letterhead on your forms
- Alternative character sets or symbols (such as a copyright symbol)
- The appearance of a physical form (by adding vertical and horizontal lines that act as boundaries on the form or between positions on an invoice)
- IPDS bar code commands

DATE

Date. Use this field-level keyword to display the current job date.

DATFMT

Date Format. Use this field-level keyword to specify the format of a date field.

DATSEP

Date Separator. Use this field-level keyword to specify the separator character for a date field.

DFNCHR

Define Character. The DFNCHR keyword allows you to define characters of your own design at the file or record level for the 5224 Printer and 5225 Printer. With this keyword you can specify DFNCHR more than once at the file or record level, or as many as 50 characters each time you specify DFNCHR.

DFT

Default. Use the DFT keyword to specify a constant value for constant (unnamed) fields.

DLTEDT

Delete Edit. Use this field-level keyword to specify that the OS/400 program is to ignore any edit code or edit word keywords specified for the referenced field. If a field description is referred to from a database file, DLTEDT prevents certain information from being referenced.

DOCIDXTAG

Document Index Tag. Use this record-level keyword to create an indexing tag in the document for use by presentation systems or postprocessor applications.

DRAWER

Drawer. Use this record-level keyword to specify the drawer from which noncontinuous forms will be selected.

DTASTMCMD

Data Stream Command. Use record-level or field-level keyword to store a data stream command in a MODCA NOP structured field in the data stream of a spooled file. This information can be used to determine how to process a record or field on a particular page of a spooled file. DTASTMCMD keyword is only valid for printer files that have the device type specified as *AFPDS.

DUPLEX

Duplex. Use this keyword to specify whether printing will be on one or both sides of the paper.

EDTCDE

Edit Code. Use this keyword to edit output-capable numeric fields.

An edit code is a letter or number indicating that editing should be done according to a defined pattern before a field is displayed or printed.

EDTWRD

Edit Word. If you cannot accomplish the desired editing by using the EDTCDE keyword, specify an edit word instead.

An edit word is a user-defined word that specifies the form in which the field values are to print and clarifies the data by inserting characters, such as decimal points, commas, floating- and fixed-currency symbols, and credit balance indicators. Also use it to suppress leading zeros and to provide asterisk fill protection.

ENDPAGE

End Page. Use this record-level keyword to specify that the page should eject. Page ejection occurs after the record format is processed.

ENDPAGGRP

End Page Group. Use this record-level keyword to end a logical grouping of pages previously started with the STRPAGGRP keyword.

FLTFIXDEC

Floating-Point to Fixed Decimal. Use this field-level keyword to print a number in a floating point field in fixed decimal notation.

FLTPCN

Floating-Point Precision. Use this keyword to specify the precision of a floating-point field.

FNTCHRSET

Font Character Set. Use this record-level or field-level keyword to specify the font and point size for printing a named or constant field or fields within a record.

FONT

Font. Use the FONT keyword to specify the font ID and point size for printing a named or constant field or fields within a record. The valid values for the point size parameter are 0.1 through 999.9.

FONTNAME

Font name. Use this file-, record-, or field-level keyword to specify the TrueType font name for printing a named or constant field within a record.

FORCE

Force. Use the FORCE keyword for duplex printing to force printing to a new sheet of paper for the next record.

GDF

Graphics Data Format File. Use this record-level keyword to specify the inclusion of a graphic data file at a specific location on a page.

HIGHLIGHT

Highlight. Use this record- or field-level keyword to indicate that a field should be printed in bold letters.

IGCCDEFNT

DBCS Coded Font. Use this record- or field-level keyword to print DBCS data contained in the AFPDS.

INDARA

Indicator Area. Use this keyword to remove option indicators from the buffer (also called the record area) and place them in a 99-byte separate indicator area.

INDTXT

Indicator Text. Use this file-, record-, or field-level keyword to associate descriptive text (indicating intent or use) with a specific indicator. You can specify INDTXT once for each indicator.

INVDTAMAP

Use this record-level keyword to start a new data map. The Invoke data map keyword specifies the name of the data map in a page definition that is used to map the line data.

INVMMAP

Invoke Medium Map. Use this keyword to invoke a new medium map. The medium map is in a form definition and allows you to select or change print parameters (input drawer, page rotation, and overlays).

You can specify the medium map name as a constant or program-to-system field. To use this keyword, you must specify a form definition on the printer file. Do not specify *NONE on the form definition parameter.

LINE

Line. Use this record-level keyword to position where a line will be drawn on a page. You can specify the weight (thickness) and color of lines. Only horizontal and vertical lines can be drawn.

LPI

Lines Per Inch. Use this keyword to change lines per inch within a file. If you do not specify LPI for a record, the LPI value is set from the LPI value on the CRTPRTF, CHGPRTF, or OVRPRTF command.

MSGCON

Message Constant. Use this field-level keyword to indicate that the text for a constant field is contained in a message description. If the message description does not exist at DDS compile time, the file is not created. If you change the message description, you must create the file again.

OUTPUT BIN

Output bin. On printers capable of multiple output bins, this keyword specifies the destination of the output.

OVERLAY

Overlay. Use this record-level keyword to specify the inclusion of an overlay to be printed at a specific location on a page.

PAGNBR

Page Number. Use this field-level keyword to specify the location of an unnamed, four-digit, zoned decimal field to contain the page number.

PAGRTT

Page Rotation. Use this record-level keyword to specify the degree of text rotation, with respect to the way the printer loads the page. The PAGRTT keyword is valid for the following printers: 3130, 3812, 3816, 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900, 3912, 3916, 3935, 4028, 4312, 4317, 4324. It is also valid for the InfoPrint 20, InfoPrint 32, InfoPrint 3000, and InfoPrint 4000. If you do not specify a PAGRTT keyword for a record, the page rotation value is taken from the value that is specified for either: the Create Printer File (CRTPRTF), Change Printer File (CHGPRTF), or Override with Printer File (OVRPRTF) commands.

Note: Overlays and page segments do not rotate with the rest of the page.

PAGSEG

Page Segment. Use this record-level keyword to specify the inclusion of a page segment to be printed at a specific location on a page.

POSITION

Position. Use this field-level keyword to specify the position of a field using the units specified on the unit of measure (UOM) printer file parameter.

PRTQLTY

Print Quality. Use this record- or field-level keyword to vary the print quality within the file.

The PRTQLTY keyword is allowed only on records or fields for which a CHRSIZ or BARCODE keyword applies.

REF

Reference. Use this file-level keyword to specify the name of a file from which field descriptions are to be retrieved.

REFFLD

Referenced Field. Use this field-level keyword when referring to a field under one of these three conditions:

- The name of the referenced field is different from the name in positions 19 through 28.
- The name of the referenced field is the same as the name in positions 19 through 28, but the record format, file, or library of the referenced field is different from that specified with the REF keyword.
- The referenced field occurs in the same DDS source file as the referencing field.

SKIPA

Skip After. Use this file-, record-, or field-level keyword to specify that the printer device is to skip to a specific line number after it prints one or more lines.

SKIPB

Skip Before. Use this file-, record-, or field-level keyword to specify that the printer device is to skip to a specific line number before it prints the next line(s).

SPACEA

Use this record- or field-level keyword to specify that the printer device is to space some number of lines after it prints one or more lines.

SPACEB

Space Before. Use this record- or field-level keyword to specify that the printer device is to space some number of lines before it prints the next line or lines.

STAPLE

Use this record-level keyword to specify that a stapling operation (defined on the CORNERSTPL, EDGESTITCH, or SADLSTITCH parameters on the CRTPRTF, CHGPRTF, or OVRPRTF commands) should be done within the spool file.

STRPAGGRP

Start Page Group. Use this record-level keyword to begin a logical grouping of pages. Online viewing products can use page groups for indexing and retrieving information in the document.

TEXT

Text. Use this record- or field-level keyword to supply a text description (or comment) for the record format or field.

TIME

Time. This field-level keyword prints the current system time.

TIMFMT

Time Format. Use this field-level keyword to specify the format of a time field.

TIMSEP

Time Separator. Use this field-level keyword to specify the separator character used for a time field.

TRNSPY

Transparency. This field-level keyword prevents code points you have redefined (using the DFNCHR keyword) from being interpreted as SCS printer control commands when your program sends an output operation that prints the field you are defining.

TXTRTT

Text Rotation. Use this field-level keyword to specify the rotation of the text on a page.

UNDERLINE

Underline. Use this field-level keyword to specify that the OS/400 program is to underline the field when it is printed.

ZFOLD

Z fold. Use this record-level keyword to specify a z-fold operation, for the current sheet, after the printer has completed printing.

Close processing

Part **3** of the application program performs the close operations of the application program.

When the application program has finished the output processing part of the application program, it performs a close operation on all the files it opened during the open processing of the application program.

The application program looks at the SCHEDULE parameter from the CRTPRTF command during the close portion of the application program. For more information on how the SCHEDULE parameter is used to control when the spooled file is made available to a printer writer, go to "Using the SCHEDULE parameter" on page 76.

Output from the example application programs

Following are examples of the output from the mailing label application program. Notice that the first example has no underlining or boldface text. Example 1 is produced using a program-described printer file and example 2 (with the underlining and highlighting) is produced using an externally-described printer file (DDS).

Using a program-described printer file

Bonnie Fohrman
Box 123
RR 1
Chatfield, IA   55924

Cindy Olson
502 9th Ave.
RR2
St. George Cove, IN   46043

Lynn Smith
808 5th St.
Peru, IN   46234

Michael Jones
Box 215
Preston, MN   55923

Pat Edwards
123 River Road
St. Charles, MN   55945

Using an externally described printer file (using DDS)

Bonnie Fohrman
Box 123
RR 1
Chatfield, IA   55924

Cindy Olson
502 9th Ave.
RR2
St. George Cove, IN   46043

Lynn Smith
808 5th St.
Peru, IN   46234

Michael Jones

Box 215
Preston, MN   55923

Pat Edwards
123 River Road
St. Charles, MN   55945

Overriding printer files

Note:

> Overriding files (printer files, display files, diskette files, database files, and tape files) can be done by commands, from CL programs, or from HLL programs. Overrides can be called from different levels (programs calling another program).
>
> This guide discusses overriding printer files.
>
> To read about overriding one file type with another, programs calling overrides from other programs (multiple call levels), and file redirection, see the File Management topic in the iSeries Information Center.

Overrides are used to temporarily specify a different printer file or temporarily change some of the attributes of a file. An override is only active for the current sign-on session. As soon as you end your session or use the Delete Override (DLTOVR) command, the override is no longer active.

Override commands may be entered interactively from a display station or as part of a batch job. They may be included in a control language (CL) program, or they may be issued from other programs via a call to the program QCMDEXC. Regardless of how they are issued, overrides remain in effect only for the job, program, or sign-on session in which they are issued. Overrides have no effect on other jobs that may be running at the same time.

Overrides are particularly useful for making minor changes to the way a program functions or for selecting the data on which it operates, without having to recompile the program. Their principal value is in allowing you to use general purpose programs in a wider variety of circumstances. Examples of items where overrides may be used are:

- Changing the name of the file to be processed
- Indicating whether output is to be spooled
- Changing printer characteristics such as lines per inch and number of copies

It is also possible to use overrides to direct data input or data that is output to a device of a different type. For example, sending data that is intended for a diskette to a printer instead. This use of overrides requires somewhat more foresight than the override applications listed above. The program must be able to accommodate the different characteristics of the two devices involved. For information about the special considerations that are required for overrides that change the file type or redirect files, see the Distributed Data Management topic in the iSeries Information Center.

Files are associated with an application program by the file names specified in the program when it is created. You can override these file names or attributes of a specified file when you compile a program or run a program. The system supplies

three override functions: applying overrides, deleting overrides, and displaying overrides. You can process override functions for files using the following CL commands:

OVRPRTF

Override with Printer File: Overrides (replaces) the printer file named in the program, overrides certain parameters of a printer file that is used by the program, or overrides the file and certain parameters of the file to be processed.

DLTOVR

Delete Override: Deletes one or more file overrides that were previously specified in a call level.

DSPOVR

Display Override: Displays file overrides at any active call level for a job.

You can use overrides to change most, but not all, of the file attributes that are specified when the file is created. In some cases, you can specify attributes in overrides that are not part of the original file definition. Refer to the command descriptions available in the online help for more information.

Overriding a file is different from changing a file in that an override does not permanently change the attributes of a file. For example, if you override the number of copies specified in a printer file by requesting six copies instead of two, the file description for the printer file still specifies two copies, but six copies are printed. The override command tells the system which file to open and what its file attributes are.

Applying overrides

There are three different types of file overrides. These are:

- Overriding file attributes

  For example, you could temporarily change the number of copies you want printed from one to three by overriding the value specified on the COPIES parameter.

- Overriding file names or types

  For example, you could temporarily change which printer file the program uses.

- Overriding file names and file attributes of the new file

  For example, you could temporarily change the printer file used and also temporarily change some of the values specified in the new printer file.

Overriding file attributes

The simplest form of overriding a file is to override some attributes of the file.

For example, assume that you create a printer file named OUTPUT with these attributes:

- Page size: 66 by 132
- Lines per inch: 6
- Copies of printed output: 2
- Pages for file separators: 2
- Overflow line number: 55

The Create Printer File (CRTPRTF) command used to create this file looks like this:

```
CRTPRTF FILE(QGPL/OUTPUT) SPOOL(*YES)
PAGESIZE(66 132) LPI(6)
COPIES(2)  FILESEP(2)  OVRFLW(55)
```

The printer file named OUTPUT is specified in your application program with an
overflow line number of 58. However, before you run the application program, you
want to change the number of copies of printed output to 3 and the overflow line
to 60. The override command looks like this:

```
OVRPRTF  FILE(OUTPUT)  COPIES(3)  OVRFLW(60)
```

When you call the application program, three copies of the output are printed.

When the application program opens the file, the file overrides, program-specified
attributes, and file attributes are merged to form the open data path (ODP) which
is used to manage the file during the running of the program. File overrides have
precedence over program-specified attributes. Program-specified attributes have
precedence over file-specified attributes. In this example, when the file is opened
and output operations are performed, spooled output will be produced with a
page size of 66 by 132, six lines per inch, three copies, two file separator pages,
and overflow at 60 lines.

The following chart explains this example:

Program A

```
.
.
.
Open OUTPUT
.
.
.
```

File OUTPUT

```
SPOOL(*YES)
PAGESIZE(66 132)
LPI(6)
COPIES(2)
FILESEP(2)
OVRFLW(55)
```

Program-Specified
Attributes

```
OVRFLW(58)
```

Override Command

```
COPIES(3)
OVERFLW(60)
```

Open Data Path

```
SPOOL(*YES)
PAGESIZE(66 132)
LPI(6)
COPIES(3)
FILESEP(2)
OVRFLW(60)
```
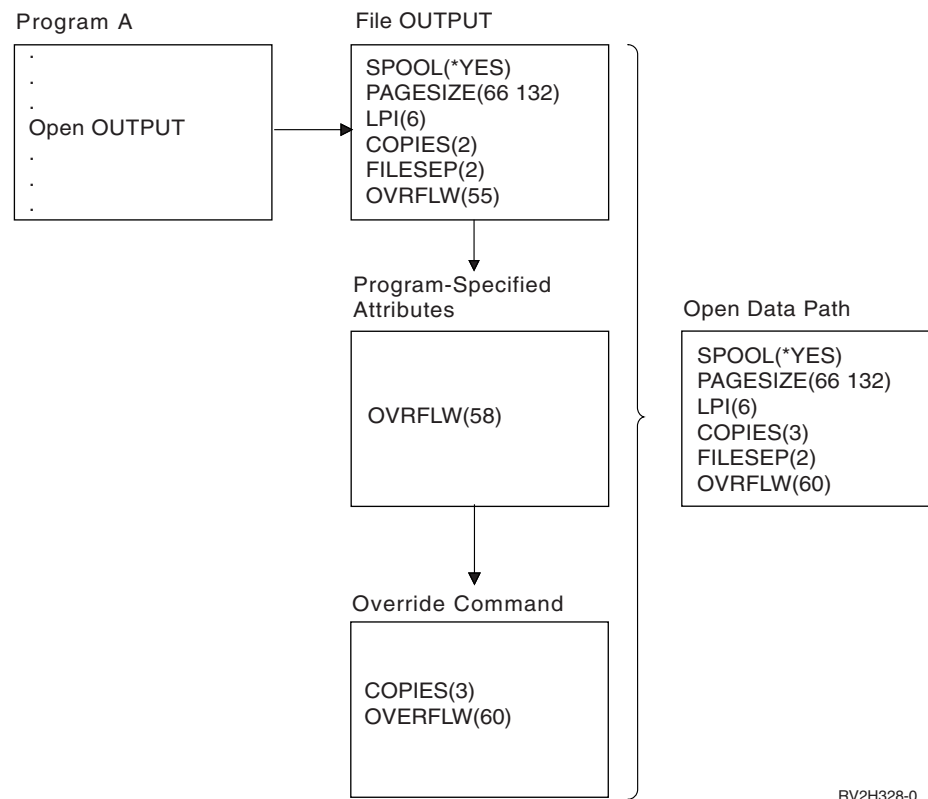
RV2H328-0

Overriding file names or types

Another simple form of overriding a file is to change which file is used by the
program. This may be useful for files that have been moved or renamed after the
program has been compiled. For example, you want the output from your
application program to be printed using the printer file named REPORTS instead
of the printer file named OUTPUT (OUTPUT is specified in the application
program). Before you run the program, enter the following:

```
OVRPRTF FILE(OUTPUT) TOFILE(REPORTS)
```

The file REPORTS must have been created (CRTPRTF command) before it can be used.

If you want to override to a different type of file, you use the override command for the new type of file. For example, if you are overriding a diskette file with a printer file, use the Override with Printer File (OVRPRTF) command.

Overriding file names or types and file attributes of the new file

This form of overriding files is simply a combination of overriding file attributes and overriding file names or types. With this form of override, you can override the file that is to be used in a program and you can also override the attributes of the overriding file. For example, you want the output from your application program to be printed using printer file REPORTS instead of printer file OUTPUT (OUTPUT is specified in the application program). In addition to having the application program use printer file REPORTS, you want to override the number of printed copies to three. Assume printer file REPORTS was created with the following command:

```
CRTPRTF FILE(REPORTS) SPOOL(*YES)
PAGESIZE (68 132) LPI(8)
OVRFLW(60) COPIES(2) FILESEP(1)
```

Before you run the program, type the following command:

```
OVRPRTF FILE(OUTPUT) TOFILE(REPORTS) COPIES(3)
```

Then call the application program, and three copies of the output are printed using the REPORTS printer file.

Note that this is <u>not</u> equal to the following two override commands:

Override 1
```
        OVRPRTF    FILE(OUTPUT)    TOFILE(REPORTS)
```

Override 2
```
        OVRPRTF    FILE(REPORTS)    COPIES(3)
```

Only one override is applied for each call level for an open of a particular file, so if you want to override the file that is used by the program and also override the attributes of the overriding file from one call level, you must use a single command. If two overrides are used, the first override will cause the output to be printed using the REPORTS printer file, but the second override will be ignored.

CL program overrides

If a CL program overrides a file and then calls a high-level language program, the override remains in effect for the high-level language program. However, if a high-level language program calls a CL program that overrides a file, the override is deleted automatically when control returns to the high-level language program.

High-level language program:

```
CALL  CLPGM1
```

  CL program:

```
  OVRPRTF  FILE(PRTF1)  TOFILE(MSTOUT)

  .

  .
```

```
      .
   ENDPGM
```

High-level language program:

```
OPEN  PRTF1
```

The file opened is PRTF1, not MSTOUT. This is because the override in the CL
program is deleted when the CL program ends.

Securing printer files

You may want to prevent the person or program that calls your program from
changing the printer file names or parameters you have specified.

You can prevent additional printer file overrides by specifying SECURE(*YES) on
the printer file override command for each printer file you want to protect from
overrides.

Using a generic override for printer files

The OVRPRTF command allows you to have one override for all the printer files
in your job with the same set of values. Without the generic override, you would
have to do a separate override for each of the printer files.

By specifying *PRTF as the file name on the OVRPRTF command, you can apply
one override to all printer files

The override specified on the OVRPRTF command with *PRTF is applied if there is
no other override for the printer file name at the same call level. The following
example shows how *PRTF works:

Override 1
```
     OVRPRTF FILE(OUTPUT) COPIES(6) LPI(6)
```


Override 2
```
     OVRPRTF FILE(*PRTF) COPIES(1) LPI(8)

     CALL PGM(X)
```

When program X opens the file named OUTPUT, the opened file has the following
attributes:

COPIES(6)
> From Override 1

LPI(6)   From Override 1

When program X opens the file named PRTOUT, the opened file has the following
attributes:

COPIES(1)
> From Override 2

LPI(8)   From Override 2

Deleting overrides

If you want to delete an override before the program has completed running, you
can use the Delete Override (DLTOVR) command. This command deletes only

overrides that are active in the call level in which the command is entered. To identify an override, use the printer file name specified on the FILE parameter of the override command.

Following is an example that shows an override of a printer file (PRTF1) with a different printer file (PRTF2). On the second line of the example is the Delete Override (DLTOVR) command that is used to delete the override. The printer file named PRTF1 will be used to process the output from the application program.

```
OVRPRTF FILE(PRTF1)  TOFILE(PRTF2)
DLTOVR FILE(PRTF1)
```

You can delete all printer file overrides at this call level by specifying *ALL for the FILE parameter.

Following is an example that shows two printer file overrides, PRTC and PRT3. The override is changing the COPIES parameter value in both cases. The third line of the example shows the Delete Override (DLTOVR) command with the FILE parameter value set to *ALL. This means the overrides to PRTC and PRT3 are deleted.

```
OVRPRTF FILE(PRTC)  COPIES(2)
OVRPRTF FILE(PRT3)  COPIES(4)
DLTOVR FILE(*ALL)
```

Displaying overrides

You can use the Display Override (DSPOVR) command to display printer file overrides. You can display all printer file overrides or overrides for a specific printer file.

To display any overrides to the printer file PRTF1, type:

```
DSPOVR FILE(PRTF1)
```

To display all printer file overrides, type:

```
DSPOVR  FILE(*ALL)
```

However, when the FILE parameter has a value of *ALL, the system presents a display with all the active overrides to all the file types supported on the system. Using FILE(*ALL) on the DSPOVR command is an indirect way to learn what overrides are active for any printer files; a more direct way is to issue a DSPOVR command for each printer file.

Changing printer files

You can change the parameters of a printer file by using the Change Printer File (CHGPRTF) command.

Using the CHGPRTF command makes the specified changes permanent for the current active session and for all future sessions. You can still apply overrides to a printer file that has been changed. However, the override is only active for the current session.

Let's assume that you have a mailing label program that uses a printer file named LABELPR3. You now want to have two sets of mailing labels every time the mailing label program is run and you want the spooled file to go to the output queue named LABELS. The CL command would look like:

```
CHGPRTF FILE(LABELPR3) COPIES(2) OUTQ(LABELS)
```

When you enter this CHGPRTF command, it becomes effective immediately.

When any application program that uses printer file LABELPR3 is run, the spooled file will go to output queue LABELS and two copies of the mailing labels will be printed. For more information on the CHGPRTF command, see the CL Reference topic in the Information Center.

Using printer file parameters

The following sections discuss individual printer file parameters in more detail to help you better understand how to use them.

Using the device type (DEVTYPE) parameter

The device type (DEVTYPE) parameter specifies the type of data stream created for a printer file. This parameter indicates whether the resulting data stream should be an intelligent printer data stream (*IPDS), an SNA character stream (*SCS), an ASCII data stream (*USERASCII), an Advanced Function Printing data stream (*AFPDS), line data (*LINE), or mixed data (*AFPDSLINE).

An AFP data stream can be created from several sources:
- System/390
- PrintManager*
- AFP utilities for OS/400
- Using the virtual print function with the iSeries Access for Windows licensed program

However, to use the functions provided by many of the printer file parameters that are supported for AFPDS, the spooled file must be created with a printer file that specifies a device type of *AFPDS. These parameters include FRONTMGN, BACKMGN, FRONTOVL, BACKOVL, FNTCHRSET, CDEFNT, and IGCCDEFNT. In cases where the iSeries server does not actually generate the AFP data stream (the list above) either a printer file is not used or the DEVTYPE parameter for the printer file is ignored.

Throughout this manual, AFPDS output that is created on the iSeries server with a printer file specifying DEVTYPE(*AFPDS) is referred to as *AFPDS created on the iSeries server.

If DEVTYPE(*USERASCII) is specified, the user is responsible for the content of the entire data stream (such as PPDS for the 4019 printer). The iSeries server will not send any formatting commands that correspond to the spooled file attributes. For example, the system sends initial formatting commands to the printer that set up the page size, lines per inch, characters per inch, and font for spooled files with DEVTYPE(*SCS) or DEVTYPE(*IPDS). These commands are not sent for spooled files with DEVTYPE(*USERASCII). Instead, the system sends the contents of the spooled file without adding any formatting commands.

This spooled file can then be sent to an ASCII printer attached to an iSeries server. It is recommended that you not use file or job separators when you specify DEVTYPE(*USERASCII).

Note: Do not put hex 03 transparency commands in the data stream. Instead, put only ASCII commands the target printer can understand. The iSeries server inserts the hex 03 commands with the correct lengths; thus, no EBCDIC to ASCII translation occurs.

Using the SCHEDULE parameter

The SCHEDULE parameter can be specified with one of three values to control when the spooled file is available for a writer to produce the file.

*FILEEND

Specifies that the spooled file is made available to the writer when the file is closed

*JOBEND

Specifies that the spooled file is made available to the writer when the job that created the spooled file is ended

*IMMED

Specifies that the spooled file is made available to the writer when the file is opened

*JOBEND considerations

The SCHEDULE(*JOBEND) spooled files of a job are grouped together on their output queues when the job completes. All SCHEDULE(*JOBEND) spooled files of the same job that are grouped together on a queue are produced together by the writer. Another spooled file can be added to the top of the queue while the writer is producing a SCHEDULE(*JOBEND) spooled file. After a writer produces one file of a job that is SCHEDULE(*JOBEND), it checks the following file on the queue. If this file is from the same job and is also SCHEDULE(*JOBEND), the writer produces it next. However, if the file is from a different job or is not SCHEDULE(*JOBEND), the first file on the queue is produced next.

If you want your SCHEDULE(*JOBEND) spooled files grouped together on a SEQ(*FIFO) output queue, you must be careful not to separate the spooled files. File operations such as HLDSPLF, CHGSPLFA, and RLSSPLF are performed one at a time (even from a Work with Output Queue (WRKOUTQ) display). If operations to other files on the queue are done at the same time, your spooled files can be separated on a SEQ(*FIFO) queue. If your SCHEDULE(*JOBEND) spooled files separate, you can regroup them by changing their output priority with the Change Job (CHGJOB) command.

*IMMED considerations

When a writer is producing a SCHEDULE(*IMMED) spooled file, it may catch up to the program producing the output. When this happens, the writer must wait for the program to produce more output. Because of this, you should be careful using *IMMED for the schedule option. When this happens, the writer cannot process other spooled files. Moreover, the device cannot be used for any other work.

Normally, spooled files that are created with SCHEDULE(*IMMED) specified are assigned a smaller internal buffer than spooled files that are created with SCHEDULE(*JOBEND) or SCHEDULE(*FILEEND) specified. Using SCHEDULE(*IMMED) to assign a smaller internal buffer may allow the spooled data to be produced sooner, but can also adversely affect performance because more disk operations are needed for the same amount of spooled data.

A large internal buffer is always used for spooled print files that use certain special device requirements. Special device requirements include:

- Defined characters
- Graphics 4214
- Graphics 4234
- Graphics 522x

Changing the SCHEDULE parameter of a spooled file with the CHGSPLFA command does not affect the internal buffers used for that file.

Using the OUTPTY parameter

Once a spooled file is available to a writer, the OUTPTY parameter determines the order in which the files are produced. The OUTPTY parameter supports the value *JOB (use the default output priority for the job) and a range of values from 1 to 9. All available files that have an output priority of 1 are positioned at the top of the output queue and are the first files to be printed. Next are the priority 2 files and so on. By selecting the appropriate output priority for spooled files, you may ensure that the spooled files needed right away will be printed first.

The priority of a spooled file when it is created is set from the printer file. Use the Create Printer File, Change Printer File, or Override with Printer File commands to set the OUTPTY parameter to the desired value before the file is opened. After the file is opened, the output priority of the spooled file can be changed by using the CHGSPLFA command.

Using the align (ALIGN) parameter

The align parameter on the Start Printer Writer (STRPRTWTR) and Create Printer File (CRTPRTF) commands influences how the iSeries server issues messages to check forms alignment in printers before they start to print.

If the value on the STRPRTWTR align parameter is *WTR, the printer writer keeps track of spooled files that are to be printed and issues a forms alignment message whenever it determines that forms alignment is needed.

If this value is *FIRST, the forms alignment message is issued only for the first spooled file printed.

The following cause the printer writer to issue a forms alignment message:
- Delete or hold spooled file command (option *IMMED) issued against a spooled file with a status of WTR
- Printer writer or spooled file restarted
- Previous spooled file was created with ASCII data (virtual print) and current spooled file was not
- Forms length of previous spooled file is different than current spooled file
- File being printed is the first file to print after a printer writer has been started
- Forms type is changed (after a G reply to a CPA3394 or CPA3395 message)

  Note: A B reply to these messages allows you to skip or not receive the alignment message.
- Incorrect control characters have been detected on a work station printer in the previous file
- The Cancel key on a work station printer has been pressed while the previous file is being printed

- A C (Cancel) reply has been given to the inquiry message for unprintable characters detected on a work station printer in a previous file
- An H (Hold) reply has been given to some inquiry messages on a work station printer
- A printer writer has recovered from a communications failure on a remote work station printer

If you do not want the printer writer to control the issuing of forms alignment messages, you can specify *FILE as the value for the align parameter of the STRPRTWTR command and *YES on the align parameter of the printer file used with the application that produces the spooled file.

If the value on the STRPRTWTR align parameter is *FILE, the printer writer looks at the attributes of the spooled file created by the application program. Specifically, the printer writer looks at the align parameter value of the spooled file attributes to determine if it should send a forms alignment check.

If the align parameter value of the spooled file attributes is *YES, the printer writer sends a message to check forms alignment on the target printer.

If the align parameter value of the spooled file attributes is *NO, the printer writer does not send any messages to check forms alignment on the target printer.

If you want to avoid any forms alignment messages, you can specify *FILE for the align parameter on the Start Printer Writer (STRPRTWTR) command and *NO as the align parameter value of the printer file. With this combination of values, the iSeries server does not send any messages to check alignment.

Using the page rotation (PAGRTT) parameter

The PAGRTT parameter controls the rotation of text on the page. When *AUTO or *DEVD is specified for the PAGRTT parameter, the system determines the orientation of the printed page. For example, if the output is too wide, the page is rotated 90 degrees.

The system automatically senses when the output is too large to fit on the paper that is loaded in the printer for the following: 3831, 3835, 3900, InfoPrint 3000, and InfoPrint 4000 printers. It also senses whether the paper is larger than 8 1/2 inches by 14 inches. In the information below, the expression "too large to fit on the form" refers to a page size that is defined in the printer file parameters. If both the length and width exceed 8.5 inches, or if the length or width is greater than 14 inches, the information is too large for the form. For example, a page defined to be 13.2 inches-wide and 11 inches long is too large for a form that is 8.5 inches-wide and 14 inches-long. This would be a candidate for computer output reduction (COR). This applies for printers other than the 3831, 3835, 3900, InfoPrint 3000, and InfoPrint 4000.

If PAGRTT = *AUTO
- If the output is too large to fit on the form, computer output reduction is performed automatically.
- If the page is wider than it is long, the page is rotated 90 degrees. If it is not wider than it is long, the page is not rotated.

- *AUTO is only supported for spooled files with a device type (DEVTYPE) of *SCS, *IPDS, or *AFPDS that is created on an iSeries server. It is not supported for files with a device type of *USERASCII, *LINE, *AFPDSLINE, or *AFPDS that is not created on an iSeries server.

If PAGRTT = *DEVD

- If the output is too large to fit on the form, computer output reduction is performed automatically.
- If the page is wider than it is long, the page is rotated 90 degrees. If it is not wider than it is long, the page is not rotated.
- For computer output reduction printing, the PRTQLTY (print quality) parameter value must be either *DRAFT or *DEVD. If the PRTQLTY parameter value is *STD or *NLQ, spooled files are printed without computer output reduction (COR) and without page rotation.
- When using the PAGRTT parameter, do not specify *DEVD for the FONT parameter value. When FONT(*DEVD) is used, the system cannot determine the exact page width; therefore, page positioning may not be as intended.

Note: When the PAGRTT parameter value is *AUTO or *DEVD, computer output reduction (COR) is not provided if the spooled file has any of the *Device Requirements* fields set to Y. Use the Work with Spooled Files (WRKSPLF) command and select option 8=Attributes to view the device requirements for any spooled file.

If PAGRTT = 0, 90, 180, or 270 degrees

When these values (0, 90, 180, or 270) are specified for the PAGRTT parameter, the page size (PAGESIZE) parameter values are not automatically changed. You must specify the PAGESIZE parameter values with reference to the way the data is printed on the page. For example, using forms that are 8.5 inches wide by 11 inches long and printing at 6 lines per inch with a 10-pitch font:

- Specify PAGESIZE(66 85) with PAGRTT(0) or PAGRTT(180).

  The page reads top to bottom with the 8.5-inch side at the top (portrait orientation).

- Specify PAGESIZE(51 110) with PAGRTT(90) or PAGRTT(270).

  The page reads top to bottom with the 11-inch side at the top (landscape orientation).

If PAGRTT = *COR

- Output is rotated 90 degrees.
- Page size is set to 11 by 8.5 inches.
- Font substitution occurs as follows:
  - 12-pitch fonts are replaced with 15-pitch fonts.
  - 15-pitch fonts are replaced with 20-pitch fonts.
  - All other fonts are replaced with a 13.3-pitch font (with the exception of the 4028 printer, which uses a 15-pitch font).

Note: When the PAGRTT parameter value is *COR, computer output reduction is not provided if the spooled file is *AFPDS, was created on the iSeries server, and has any of the *Device Requirements* fields set to Y.

Use the Work with Spooled Files (WRKSPLF) command and select option 8=Attributes to view the device requirements for any spooled file. In addition, *COR is not supported for spooled files with a device type of:

- – *USERASCII
  - – *LINE (if the spooled file is sent to a printer configured with AFP(*YES))
  - – *AFPDSLINE
  - – *AFPDS (if the spooled file was not created on the iSeries server)

Page rotation on the 3831, 3835, 3900, InfoPrint 3000, and InfoPrint 4000 printers

The 3831, 3835, 3900, InfoPrint 3000, and InfoPrint 4000 printers are continuous-forms printers that can accept page rotation commands. Paper that is wider than it is long has output that is printed at 90-degree rotation. The paper rotation is in a counter-clockwise direction. Normally output rotation is in a clockwise direction. See the diagram below for an example of how printed output appears on these printers when rotation occurs.

Notes:

1. The corner with the **X** is the physical paper origin. It is the left edge of the narrow side of the page.

2. The circle with the plus (+) sign in it designates the logical paper origin.

RV3H005-0
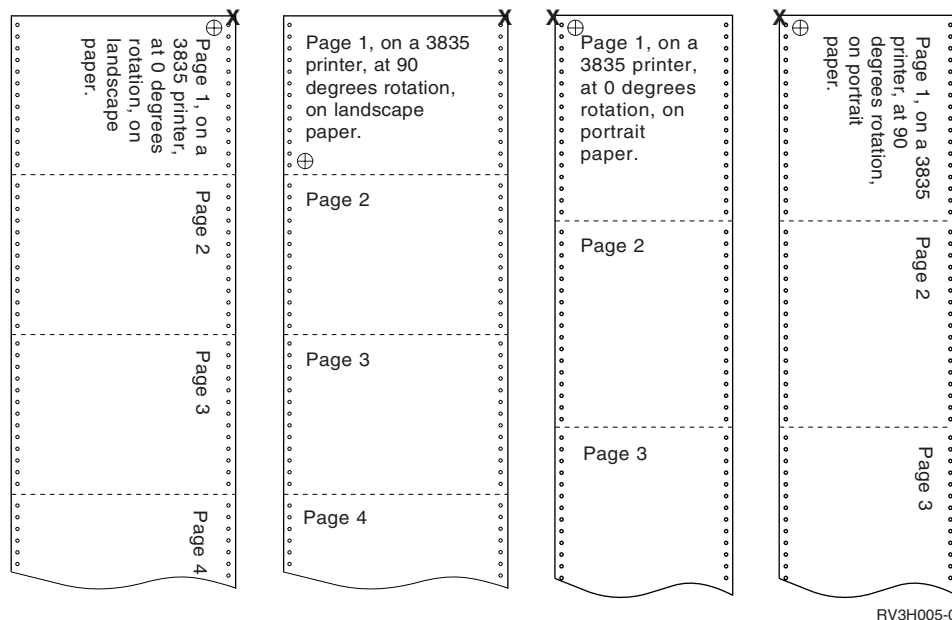
Page rotation and duplex printing

The examples below show how printed output will appear when combinations of duplex and page rotation are used. The dots on each page represent holes punched in the paper.

PAGRTT (0)
DUPLEX (*YES)

Front Side   Back Side

PAGRTT (0)
DUPLEX (*TUMBLE)

Front Side   Back Side
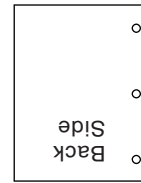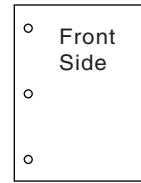
PAGRTT (90)
DUPLEX (*YES)

Front Side   Back Side
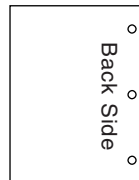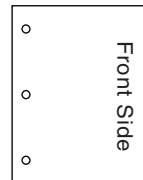
PAGRTT (90)
DUPLEX (*TUMBLE)

Front Side   Back Side
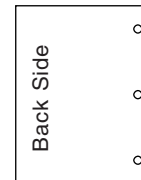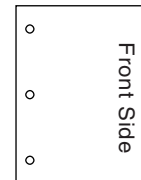
RV2H310-1

Note: A page segment or overlay included using the DDS PAGSEG or OVL keyword is not rotated with the rest of the printed output. This is also true of an overlay (front or back) specified on the printer file.

You must specify the degree of rotation when the page segment or overlay is created. Advanced Function Printing Utilities/400 can be used to create overlays and page segments.

Using the MULTIUP(1, 2, 3, or 4) and REDUCE(*TEXT) parameters

Note: The value 3 is not allowed for MULTIUP when REDUCE(*TEXT) is specified.

The MULTIUP function can be software or hardware controlled. The software MULTIUP function is selected by specifying REDUCE(*TEXT) on the printer file. The hardware MULTIUP function is selected by specifying REDUCE(*NONE) on the printer file.

The information presented here is dependent on specifying a REDUCE parameter value of *TEXT.

For information on MULTIUP when the REDUCE parameter value is *NONE, see "Using the MULTIUP(1, 2, 3, or 4) and REDUCE(*NONE) parameters" on page 86.

The MULTIUP parameter allows you to print more than one logical page of data on a piece of paper. Depending on the printer that you are using, you can print from one to eight logically formatted pages on one physical piece of paper.

Using a printer that supports duplex printing, you could have four logical pages printed on each side of the physical paper, thus reducing the number of physical pages printed from eight to one.

The rotation value specified in the PAGRTT parameter of the printer file is used to determine the rotation for MULTIUP printing. When PAGRTT(*AUTO), PAGRTT(*DEVD), or PAGRTT(*COR) is specified and the MULTIUP parameter has a value of 2 or 4, the PAGRTT value used is 0.

Note: If overlays (using the front overlay (FRONTOVL) and back overlay (BACKOVL) parameters) are included in output using MULTIUP support, the overlay applies to the whole sheet. That is, the front overlay is placed on the front side of the paper and the back overlay on the back side.

MULTIUP support for the 4224, 4230, 4234, 4247 printers

These are continuous-forms printers, and do not support rotation. MULTIUP(2) with PAGRTT(0) is the only valid combination. Below is an example of what the physical output page would look like. If the logical page width is less than or equal to 6-1/2 inches, the font is not changed. Otherwise font 223, which is 15 pitch, is used.

Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2

MULTIUP support

MULTIUP support is available on the following printers:

- 3130 3812 3816 3820 3825 3827 3828 3829 3831 3835 3900 3912 3916 3930 3935 4028 4312 4317 4324 Infoprint® 20 Infoprint 32

These are IPDS printers that support page rotation (PAGRTT) and pages per side (MULTIUP).

For all the above-described printers, the following diagrams are examples of how the output will print depending on the PAGRTT and MULTIUP values.

PAGRTT(0) or PAGRTT(180) and MULTIUP(2)

Page 1
Page 1
Page 1
Page 1
Page 1
Page 1
Page 1
Page 2
Page 2
Page 2
Page 2
Page 2
Page 2
Page 2

If the logical page width is less than or equal to 8 inches and the logical page length is less than 5 inches, the font is not changed.

If the logical page width is greater than 8 inches, or the logical page length is greater than 5 inches, the font is selected based on the following table:

Note: Any references to the 4028 printer in the following tables also apply to the 3912, 3916, 4312, 4317, 4324, InfoPrint 20, and InfoPrint 32 printers. The following footnotes apply to all of the tables about MULTIUP Font Substitution that are included here.

Notes:

1. If the target printer does not have font 230 resident, it uses font 223.
2. If the target printer is a 4028 and has font 283 (20 pitch) resident, it uses font 283. Otherwise it uses font 281 (20 pitch).
3. If the target printer is a 3130 or 3935, it will use font 416 with point size of 4 (30 pitch). If the target printer is a 4028 and does not have font 290 resident but does have font 283 resident, it will use font 283. If the target printer is a 4028 and does not have font 290 or 283 resident, it will use font 281 (20 pitch).

Table 2. MULTIUP font substitution for page rotation 0 or 180 with MULTIUP(2)

Page Width (Characters)	Page Length (Inches)	Font Used	Pitch (CPI)
1 through 123 characters	Less than or equal to 5 inches	230[1]	15
1 through 123 characters	Greater than 5 inches	254	17
124 through 139 characters	All	254	17
140 through 163 characters	All	281[2]	20[2]
Greater than or equal to 164 characters	All	290[3]	27[3]

PAGRTT(90) or PAGRTT(270) and MULTIUP(2)

Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2
Page 1	Page 2

If the logical page width is less than or equal to 5 inches and the logical page length is less than 8 inches, the font is not changed.

If the logical page width is greater than 5 inches, or the logical page length is greater than 8 inches, the font is selected based on the following table:

Table 3. MULTIUP font substitution for page rotation 90 or 270 with MULTIUP(2)

Page Width (Characters)	Page Length (Inches)	Font Used	Pitch (CPI)
1 through 73 characters	Less than or equal to 8 inches	230[1]	15
1 through 73 characters	Greater than 8 inches	254	17

Page Width (Characters)	Page Length (Inches)	Font Used	Pitch (CPI)
74 through 83 characters	All	254	17
84 through 97 characters	All	281[2]	20[2]
Greater than or equal to 98 characters	All	290[3]	27[3]

PAGRTT(0) or PAGRTT(180) and MULTIUP(4)

Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4

If the logical page width is less than or equal to 3.75 inches and the logical page length is less than 5 inches, the font is not changed.

If the logical page width is greater than 3.75 inches, or the logical page length is greater than 5 inches, the font is selected based on the following table:

Table 4. MULTIUP font substitution for page rotation 0 or 180 with MULTIUP(4)

Page Width (Characters)	Page Length (Inches)	Font Used	Pitch (CPI)
1 through 54 characters	Less than or equal to 5 inches	230*	15
1 through 54 characters	Greater than 5 inches	254	17
55 through 61 characters	All	254	17
Characters 62 through 71	All	281[2]	20[2]
Greater than or equal to 72 characters	All	290[3]	27[3]

PAGRTT(90) or PAGRTT(270) and MULTIUP(4) PAGRTT(0) or PAGRTT(180) and MULTIUP(4)

Page 1	Page 3
Page 1	Page 3
Page 1	Page 3
Page 1	Page 3

Page 2	Page 4
Page 2	Page 4
Page 2	Page 4
Page 2	Page 4

If the logical page width is less than or equal to 5 inches and the logical page length is less than 3.75 inches, the font is not changed.

If the logical page width is greater than 5 inches, or the logical page length is greater than 3.75 inches, the font is selected based on the following table:

Table 5. MULTIUP font substitution for page rotation 90 or 270 with MULTIUP(4)

Page Width (Characters)	Page Length (Inches)	Font Used	Pitch (CPI)
1 through 73 characters	Less than or equal to 5 inches	230^1	15
1 through 73 characters	Greater than 5 inches	281	20
74 through 83 characters	Less than or equal to 3.75 inches	254	17
74 through 83 characters	Greater than 3.75 inches	281	20
84 through 97 characters	All	281^2	20^2
Greater than or equal to 98 characters	All	290^3	27^3

Restrictions for MULTIUP with REDUCE(*TEXT)
- MULTIUP requires an IPDS printer. It can be configured AFP(*YES) or AFP(*NO).
- MULTIUP reduces the size of any lines drawn in the printed output resulting from the use of the DDS LINE or BOX keywords. Depending on the pel resolution of the printer being used, the lines may become too narrow to print.
- MULTIUP with REDUCE(*TEXT) is not supported for spooled files created with device type (DEVTYPE) of *LINE, *AFPDSLINE, or *AFPDS unless the AFPDS was created
- MULTIUP is ignored if the spooled file contains any of the following advanced printing functions:
  - Final form text
  - Variable fonts
  - Variable lines per inch
  - Variable drawer
  - Superscript or subscript
  - Variable character ID
  - Highlighting
  - Extended 3812 fonts
  - Graphics
  - Bar codes
  - Variable page rotation
  - PC printer emulation
  - Define characters
  - Variable characters per inch

- – Transparency commands
- – Field outlining
- – AFP resources (overlays, page segments, host resident fonts)

Using the MULTIUP(1, 2, 3, or 4) and REDUCE(*NONE) parameters

Note: Hardware MULTIUP (REDUCE(*NONE)) is only supported by some printers. The 3130 or 3935 is an example of a printer that supports hardware MULTIUP. If the target printer does not support hardware MULTIUP, the pages printed are not printed using the MULTIUP function.

To use hardware MULTIUP you can select any MULTIUP value (1 through 4), and you must specify *NONE as the value on the REDUCE parameter of the printer file. It is then up to the application to ensure the output fits in the partition (portion) of the page. The partitioning of the page is determined by the printer. This combination of MULTIUP and REDUCE parameter values also allows multiupping of data streams restricted by software multiupping. For example, *LINE and *AFPDSLINE data streams can be multiupped as can data streams which contain advanced functions and host resident fonts.

When REDUCE(*NONE)is specified, overlays and page segments apply to each partition of the multiupped page. Thus, if front overlay (FRONTOVL) and back overlay (BACKOVL) are used for a MULTIUP(2) REDUCE(*NONE) job, the FRONTOVL is printed in the first partition and BACKOVL in the second partition.

Notes:

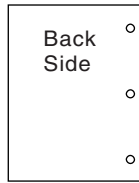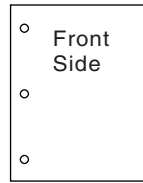1. The corner with the **X** is the physical paper origin. It is the left edge of the narrow side of the page.
2. The circle with the plus (+) sign in it designates the logical paper origin.

The following diagrams are examples of how output prints when REDUCE(*NONE) and MULTIUP(2, 3, or 4) are specified.

MULTIUP(2), REDUCE(*NONE), and PAGRTT(0)
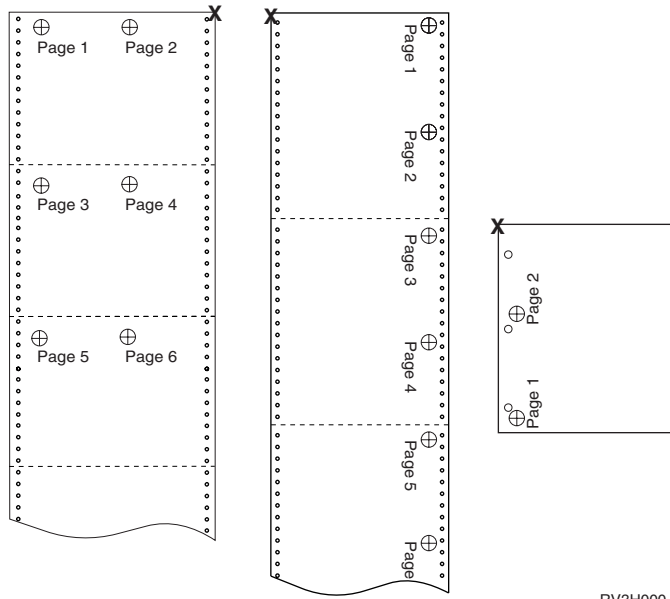
MULTIUP(3), REDUCE(*NONE), and PAGRTT(0)
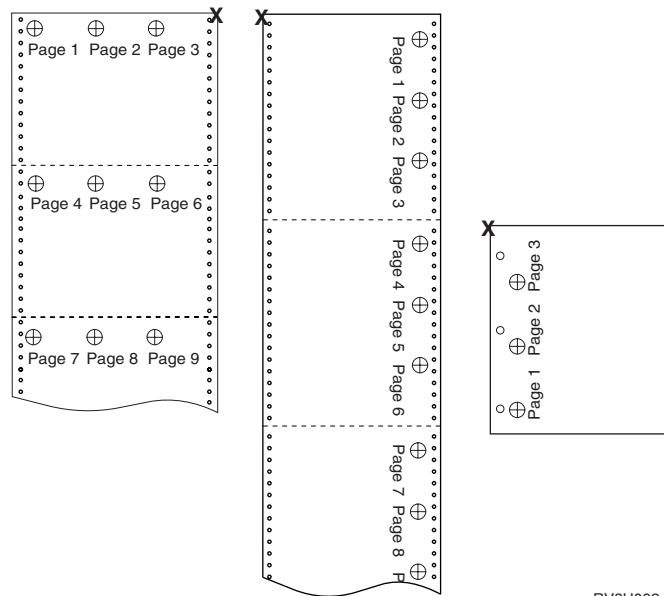
MULTIUP(4), REDUCE(*NONE), and PAGRTT(0)
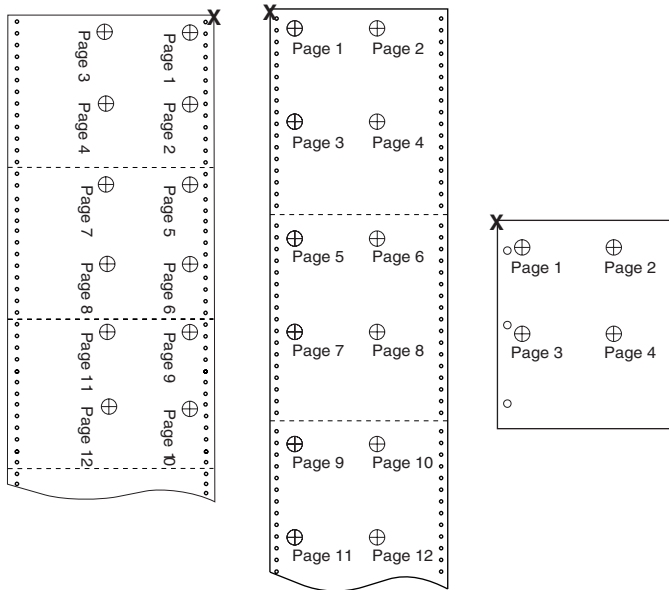
MULTIUP(4), REDUCE(*NONE), and PAGRTT(90)

Note: It is recommended *not* to use rotation with MULTIUP and REDUCE(*NONE). Following the reading direction can be difficult due to the way the data is oriented on the physical page. The example below shows how the output prints.
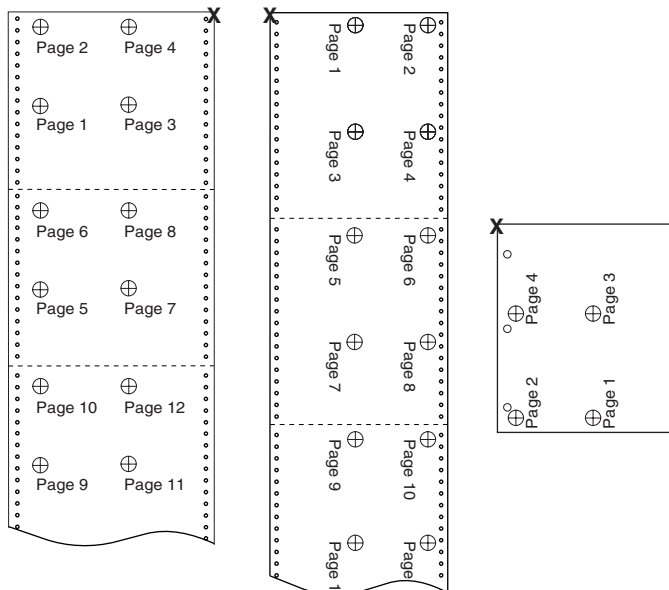
Using the fidelity (FIDELITY) parameter

For AFP functions that are not supported, the FIDELITY parameter allows the user to determine if a spooled file with advanced printing functions should continue to print on an IPDS printer configured AFP(*YES) or stop printing when an advanced printing function is encountered that is not supported. For example, if a spooled file containing bar code commands is to be printed on a 3820 printer and the FIDELITY parameter value is:

- *ABSOLUTE, the spooled file does not print because the 3820 does not support bar codes.
- *CONTENT, the spooled file prints without the bar codes.

Note: If the device description of the printer has the value for the print while converting (PRTCVT) parameter set to *YES, the spooled file can print the pages up to the point where the bar code was encountered no matter what the fidelity parameter value is.

Fidelity and other printer file parameters

If FIDELITY(*ABSOLUTE) is specified, the following list of printer file parameters must have a valid value specified for the selected printer. Otherwise, the spooled file is held.

- DRAWER
- FONT
- DUPLEX
- MULTIUP
- OUTBIN
- PAGRTT
- FRONTOVL
- BACKOVL

If FIDELITY (*CONTENT) is specified, a default value is used. For example, if DUPLEX (*YES) is requested, but the printer does not support duplex printing, the spooled file is printed on one side of the paper.

Using the overlay (FRONTOVL and BACKOVL) parameters

When an overlay is specified in a printer file, you can merge data from a spooled file onto the same piece of paper that the overlay is printed on. The FRONTOVL parameter specifies the overlay to be printed on the front side of the paper; the BACKOVL parameter specifies the overlay for the back side of the paper.

With your application program, you can use overlays that you created yourself using the Advanced Function Printing Utilities/400 licensed program, or overlays sent from System/390.

For more information about the Advanced Function Printing Utilities/400 licensed program, go to "Advanced Function Printing Utilities/400" on page 313.

The following diagram is an example of an overlay you can create using the AFP utilities. Additionally, the diagram shows how the merged spooled file data (Variable Page Data in the diagram) can be integrated into one document.

The merged document can be printed on any IPDS printer configured with AFP(*YES) in the printer's device description.

The overlays can only be merged with a spooled file that was created using a device type (DEVTYPE) of *SCS, *IPDS, or *AFPDS. The *AFPDS must have been created on an iSeries server.

Electronically Stored Form Overlay

Variable Page Data

Merge

Finished Product

RV2H309-0

Using overlays and rotation

Rotation can occur for text and overlays. The iSeries server treats text and overlays separately. That is, text is rotated using the page rotation (PAGRTT) parameter of the printer file. Overlay rotation must be determined at the time the overlay is created. Overlays are not rotated by the PAGRTT parameter.

To use the contents of the same overlay with different degrees of rotation (0 and 90 degrees), you create the overlay twice, once for 0 degrees and once for 90 degrees.

Assuming that you want the text and the overlay to be read in the same direction, the overlay offset values (down and across) on the printer file need to change depending on the degree of rotation of the text (PAGRTT parameter). That is, an overlay created with 90 degrees rotation is most often used with text that is rotated 90 degrees.

Notes:

1. Determining the overlay parameter values is made easier by focusing on a particular corner of the overlay depending on the degree of rotation specified.
2. Keep in mind that paper does not rotate as it passes through the printer.
3. The page size (PAGESIZE) values (down, across) must be viewed differently depending on the page rotation value (PAGRTT) specified. The examples show this concept by having numbers along the edge of the page.

Following are examples of how to determine the overlay parameter values when page rotation (PAGRTT) is used.

Example 1: Determining Overlay Values with Page Rotation (PAGRTT) of 0 Degrees. The following example assumes:

- Page size (PAGESIZE) parameter values are (11,9)
- Page rotation (PAGRTT) parameter value is (0)

RV2H337-1
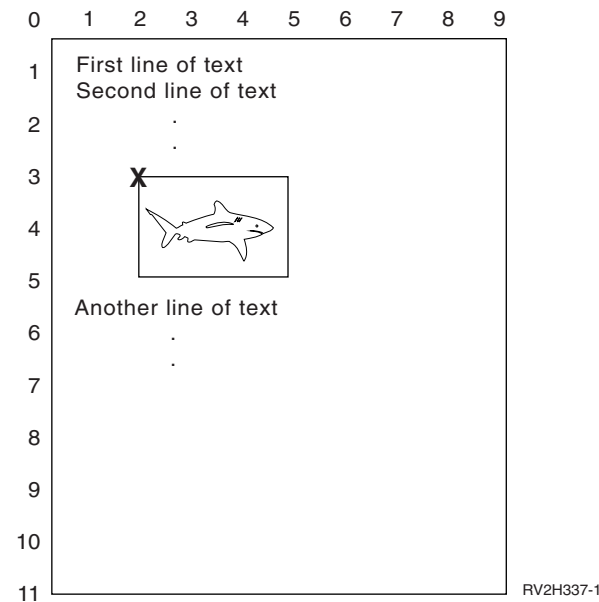
When PAGRTT is 0, the focus point for determining the overlay parameter values is the top left of the overlay (indicated by the **X** in the example).

Use the numbers beside the example page to determine the overlay down (3) and across (2) values.

Overlay parameter values are (3,2). *Example 2: Determining Overlay Values with Page Rotation (PAGRTT) of 90 Degrees*. The following example assumes:

- Page size (PAGESIZE) parameter values are (9,11)
- Page rotation (PAGRTT) parameter value is (90)

RV2H338-1
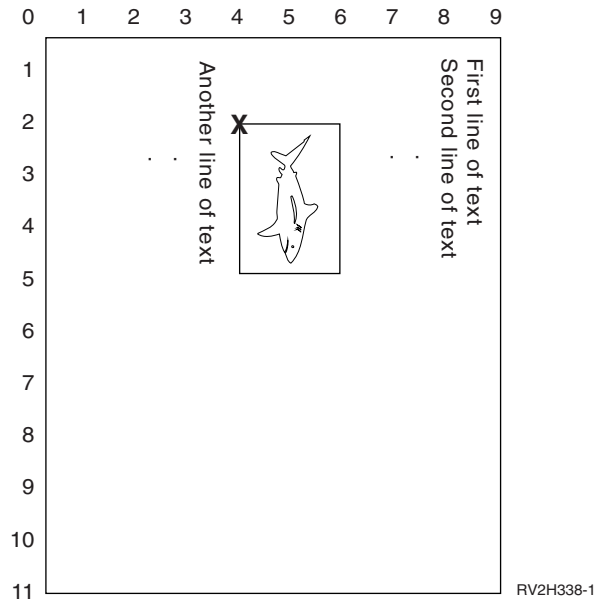
When PAGRTT is 90, the focus point for determining the overlay parameter values is the bottom left of the overlay (indicated by the **X** in the example).

Use the numbers beside the example page to determine the overlay down (2) and across (4) values.

Overlay parameter values are (2,4). *Example 3: Determining Overlay Values with Page Rotation (PAGRTT) of 180 Degrees*. The following example assumes:
- Page size (PAGESIZE) parameter values are (11,9)
- Page rotation (PAGRTT) parameter value is (180)

RV2H339-1
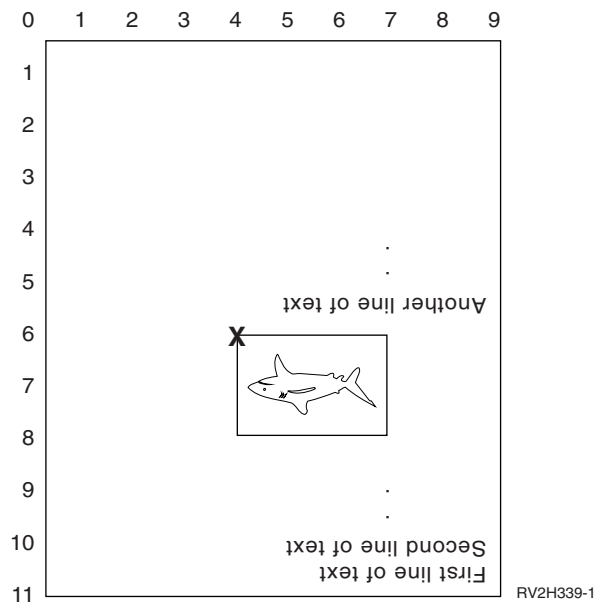
When PAGRTT is 180, the focus point for determining the overlay parameter values is the bottom right of the overlay (indicated by the **X** in the example).

Use the numbers beside the example page to determine the overlay down (6) and across (4) values.

Overlay parameter values are (6,4). *Example 4: Determining Overlay Values with Page Rotation (PAGRTT) of 270 Degrees*. The following example assumes:

- Page size (PAGESIZE) parameter values are (9,11)
- Page rotation (PAGRTT) parameter value is (270)

```
   0   1   2   3   4   5   6   7   8   9
 1
 2
 3
 4
 5
 6        X
 7        First line of text
 8        Second line of text      Another line of text
 9              .  .              .  .
10
11                                              RV2H340-1
```
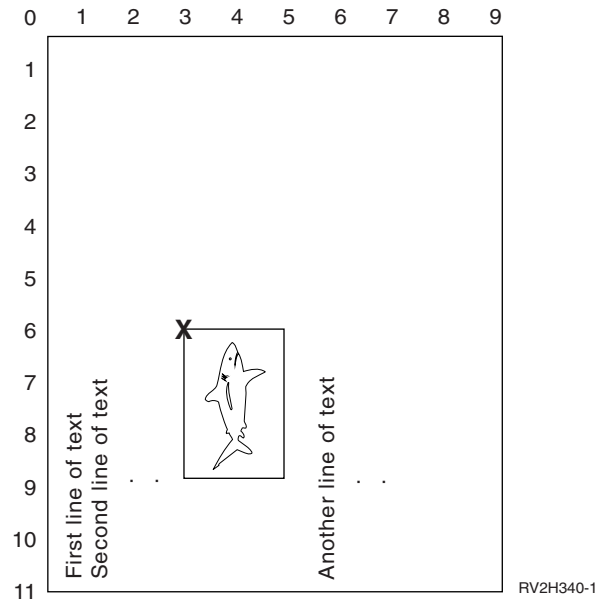
When PAGRTT is 270, the focus point for determining the overlay parameter values is the top right of the overlay (indicated by the **X** in the example).

Use the numbers beside the example page to determine the overlay down (6) and across (3) values.

Overlay parameter values are (6,3).

Using the margin (FRONTMGN and BACKMGN) parameters

To use the margin parameters, the device type (DEVTYPE) parameter on the printer file must be *AFPDS. For device types other than *AFPDS, the iSeries server calculates the margins.

Margins define the starting point of printed output on a piece of paper. The FRONTMGN parameter specifies the starting point on the front side of the paper; the BACKMGN parameter specifies the starting point on the back side of the paper.

There are two types of margins: front and back. Offset values, down and across, are used to fix the position of the margin. Across is defined as left to right. Down is defined as top to bottom.

Margins are measured in either inches or centimeters. The type of measurement is specified in the unit-of-measure (UOM) parameter on the printer file.

Using *DEVD and 0 as margin parameter values

If you have existing application programs that specify how far across and how far down to start printing, you should specify 0 (zero) or *DEVD for the margin parameter offset values.

*DEVD

> When *DEVD is specified, the no-print border (no-print border is a small area around the whole page where data will not print) is used to determine the starting printing point.
>
> For printers configured as AFP(*NO), 0 is used to determine the starting point when *DEVD is specified.

0   When 0 is specified, the top left corner of the page is used to determine the starting printing point.

The following diagram shows a no-print border. The size of the no-print border can vary from printer to printer.

Shaded Area Represents the No-Print Border

RV2H341-1
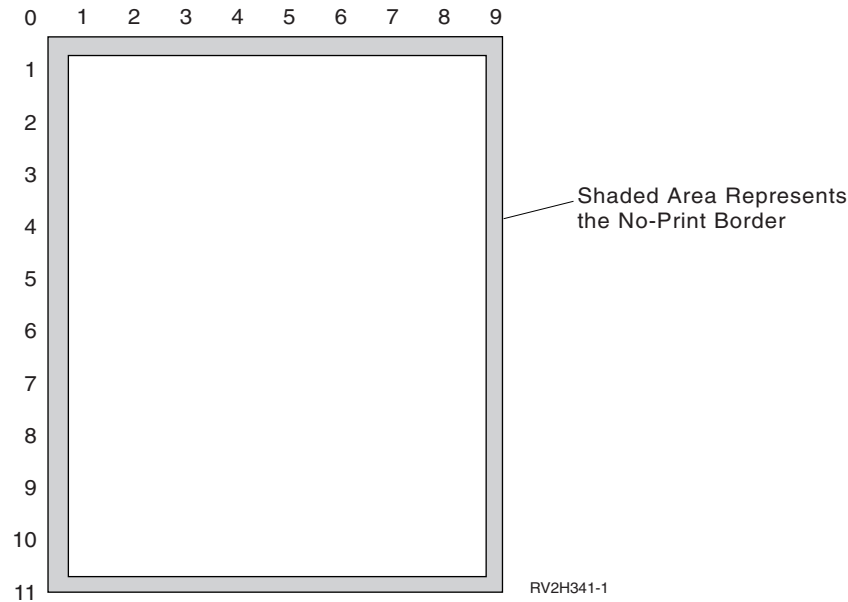
Restrictions with the margin parameters

The margin parameters (both front and back) are ignored for spooled files that have *COR specified as the value for the page rotation (PAGRTT) printer file parameter. This is because *COR assumes a half-inch margin. In addition, the margin parameters are ignored for spooled files that have a pages per side (MULTIUP) value of 2, 3, or 4.

Using margin parameters and overlays

There are two ways to position your overlays

- Specify overlays with the front or back margin (FRONTMGN or BACKMGN) parameters so that the overlays move with the text based on margin parameters.
- Change value in data area QPRTVALS so that the overlays are not affected by the front or back margin parameters.

Positioning overlays based on margin parameters: Overlays specified with the back or front overlay printer file parameter are affected by the front or back margin parameters. The following diagram shows how your output would look if

your front margin was defined as 2 units down and 2 units across and the
placement of your overlay was 4 units down and 4 units across:

```
0     1     2     3     4     5     6     7     8     9


1


                  │
2 ──────────────▶ X  Margin  of  2  units  down  and  2  units  across.


3


4                       Position  of  overlay  is  6  units
                        down  and  6  units  across.  Position
                        obtained  by  adding  the  values  of
5                       the  overlay  and  the  values  of  the
                        margin.


6 ──────────────────────────────▶ X
```
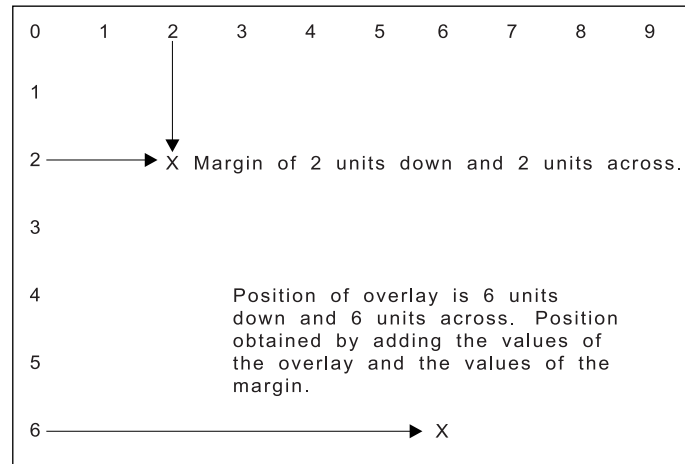
RBAFT516-0

Note: If you use margins and overlays together, the offsets have the same unit of
measure (UOM) specified: inches (*INCH) <u>or</u> centimeters (*CM), but not
inches <u>and</u> centimeters together.

Positioning overlays independently: If you do not want the overlays moved by
the values specified in the front or back margin parameters, you can specify a Y
(upper case) in position 4 of the data area QPRTVALS. The following diagram
shows you the two ways to position the overlays:

Front/back margins used   Front/back margins used
overlay position move   overlay position not affected
QPRTVALS position 4=' '(blank)   QPRTVALS position 4='Y'
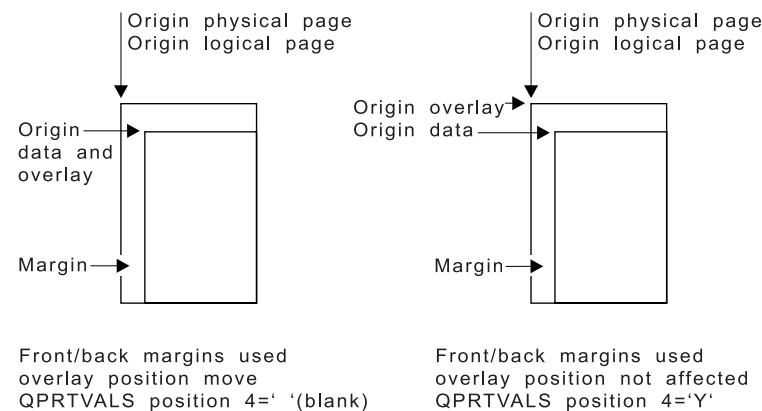
RBAFT517-0

To find out if data area QPRTVALS exists on your system, use the following
command:

`DSPDTAARA DTAARA(QUSRSYS/QPRTVALS)`

If the data area QPRTVALS exists, but position 4 is not set to ′Y′ (uppercase), use
the following command:

`CHGDTAARA DTAARA (QUSRSYS/QPRTVALS (4 1)) VALUE ('Y')`

If you do not have data area QPRTVALS on your system, you can create one by
using the following commands:

1. `CRTDTAARA DTAARA(QUSRSYS/QPRTVALS) TYPE(*CHAR) LEN(256) VALUE(' Y')`

2. `CHGOBJOWN OBJ(QUSRSYS/QPRTVALS) OBJTYPE(*DTAARA) NEWOWN(QSYS)`
   `CUROWNAUT(*SAME)`
3. `GRTOBJAUT OBJ(QUSRSYS/QPRTVALS) OBJTYPE(*DTAARA) USER(*PUBLIC) AUT(*ALL)`

Using the corner staple (CORNERSTPL) parameter

This parameter specifies the corner of the media to be used for stapling. The following diagram indicates the corners that you can specify for a printer. The corners that you can specify are device dependent. Refer to your printer's documentation for this information. Note that page rotation does not affect the staple placement.
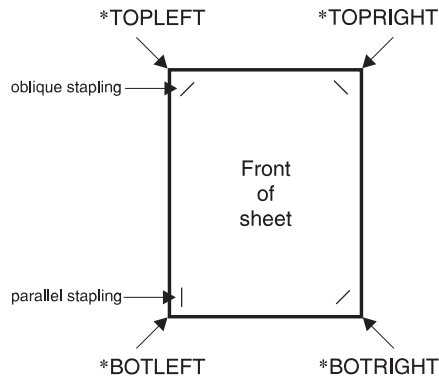
Note: The offset and angle of the staple from the selected corner are device dependent.

RV4W160-0

Using the edge stitch (EDGESTITCH) parameter

This parameter specifies where staples are to be placed along the finishing margin of the media. You need to check your printer documentation to determine which elements, and values of elements, are supported. Where your printer does not support any of the values for a specific element, specify a value of *DEVD for the element.

Note: The finishing margin is an invisible line along which finishing operations, such as edge stitching, are done. You specify the position of the finishing margin, relative to the physical edge, in the reference edge offset element of the parameter.

Element 1: Reference Edge
   Specifies which edge to use for finishing. Possible values are:

   *DEVD
       The default that is used by the device.

   *BOTTOM
       The reference edge is the bottom edge.

   *LEFT The reference edge is the left-hand edge.

   *RIGHT
       The reference edge is the right-hand edge.

   *TOP The reference edge is the top edge.

Element 2: Reference Edge Offset

Specifies the offset from the reference edge to place the edge stitching. Possible values are:

*DEVD

The default that is used by the device.

reference-edge offset

This element that is specified in centimeters (range 0 through 57.79) or inches (range 0 through 22.57).

Element 3: Number of Staples

Specifies the number of staples to use for edge stitching. Possible values are:

*DEVD

The default value for the device. This is the value that is used if *DEVD is also specified for the staple offsets value of this parameter. The system uses the default number of staples for the device when you specify *DEVD for this and for the staple offset value.

number of staples

The valid value range is from 1 to 122 staples. The number of staples is the same as the number of staple offsets specified.

Element 4: Staple Offsets

Specifies the distance between staples that are used in the edge stitching. If the staple-placement will be on the left or right edge of the paper, the first staple offset is determined by: measuring from the intersection of the finishing margin and the bottom edge of the paper to where the center of the staple will be. Subsequent staple offsets are measured from the same point (not from the previous staple). If the staple-placement will be at the top or bottom edge of the paper, the first staple offset is determined by: measuring from the intersection of the finishing margin and the left edge of the paper, to where the center of the staple will be. Subsequent staple offsets are measured from the same point (not from the previous staple). Possible values are:

*DEVD

The default staple offset for the device. If you specify a value for the Number of staples, the printer will automatically calculate the position of each staple.

staple offset

The valid value range is from 1 to 122 staple offsets. If you specify a staple offset, the Number of staples must be *DEVD. This measurement is in centimeters (range 0 through 57.79) or inches (range 0 through 22.57)

The diagram below shows the reference edge offset from the left edge with 2 staples. It uses the following values:

- Element 1: Reference edge—*LEFT
- Element 2: Reference edge offset—*DEVD
- Element 3: Number of staples—*DEVD
- Element 4: Staple offsets—*DEVD

Reference edge offset
Finishing margin
Reference edge
Front of sheet
RV4W161-1
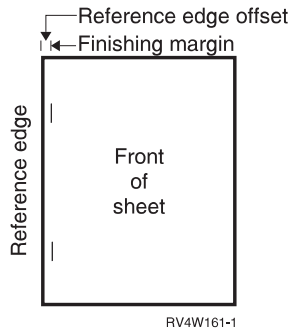
Using the saddle stitch (SADLSTITCH) parameter

This parameter specifies where to place the staples along the finishing margin of the media. In saddle stitching, the finishing margin is at the center of the media, and parallel to the reference edge. Page rotation does not affect the placement of a saddle stitch.

You need to check your printer documentation to determine which elements, and values of elements, are supported. Where your printer does not support any of the values for a specific element, specify a value of *DEVD for the element.

Element 1: Reference Edge
> Specifies which edge to use for the saddle stitch. Possible values are:
>
> ***DEVD**
>> The default that is used by the device.
>
> ***TOP** The reference edge is the top edge.
>
> ***LEFT** The reference edge is the left-hand edge.

Element 2: Number of Staples
> Specifies the number of staples to use for saddle stitching. Possible values are:
>
> ***DEVD**
>> The default value for the device. This is the value that is used if *DEVD is also specified for the staple offsets value of this parameter. When specifying *DEVD for this and for the staple offset value, the number of staples will be the default that is used by the device. If you specify one or more offsets, the number of staples is the same as the number of staple offsets specified.
>
> **number of staples**
>> The valid value range is from 1 to 122 staples.

Element 3: Staple Offsets
> Specifies the distance between staples that are used in the edge stitching. If the staple-placement will be on the left or right edge of the paper, the first staple offset is determined by: measuring from the intersection of the finishing margin and the bottom edge of the paper to where the center of the staple will be. Subsequent staple offsets are measured from the same point (not from the previous staple). If the staple-placement will be at the top or bottom edge of the paper, the first staple offset is determined by: measuring from the intersection of the finishing margin and the left edge of the paper, to where the center of the staple will be. Subsequent staple offsets are measured from the same point (not from the previous staple). Possible values are:

*DEVD

The default staple offset that is used by the device. If you specify a value for the Number of staples, the printer will automatically calculate the position of each staple.

staple offset

The valid value range is from 1 to 122 staple offsets. If a staple offset is other than *DEVD, the Number of staples must be *DEVD. This measurement is in centimeters (range 0 through 57.79) or inches (range 0 through 22.57)

Using the DBCS coded font (IGCCDEFNT) parameter

The DBCS coded font is the font that the system uses for DBCS printing on IPDS printers. This parameter is specified only for printer files with DEVTYPE (*SCS) or (*AFPDS) when printing on IPDS-capable printers configured with AFP(*YES). A point size may be specified for outline fonts. It will be ignored for raster fonts.

Transforming SCS DBCS data to AFPDS data

When transforming the SCS data stream to AFPDS, the IGCCDEFNT parameter is used to print the DBCS data. When the spooled file is created as SCS, it contains SO/SI (shift out / shift in) characters to identify the double-byte data. When the printer writer is printing a spooled file, a font change to the DBCS coded font replaces the SO in the AFPDS data stream. When the SI is encountered in the data stream, a font change is placed in the data stream to change the font back to the previous SBCS font.

Generating AFPDS data to spool

The IGCCDEFNT parameter is used when generating AFPDS. If the data stream contains double-byte data (IGCDTA(*YES)), a font substitution to the DBCS coded font takes place.

In an externally-described file (DDS), the user can use the IGCCDEFNT DDS keyword to specify a DBCS font. This font will be used to print any DBCS data encountered in that field or record. If the user has a DBCS-graphic field specified, the printer file IGCCDEFNT parameter is used unless the IGCCDEFNT DDS keyword has been specified at the record or field level.

Using the font character set (FNTCHRSET) parameter

This parameter is specified only for printer files with DEVTYPE (*AFPDS) when printing on IPDS-capable printers configured with AFP(*YES).

When using a font character set, a character set and a code page must be specified on the font character set (FNTCHRSET) parameter of the printer file being used. A point size may be specified for outline fonts. It will be ignored for raster fonts.

You cannot specify a coded font if you use the FNTCHRSET parameter on the printer file.

You can find out which font character sets and code pages come with the iSeries server by using the Work with Font Resources (WRKFNTRSC) command and specifying QFNTCPL for the library and *FNTCHRSET or *CDEPAG as the object attribute.

Font character sets and code pages are downloaded from the iSeries server to an IPDS printer when the spooled file is printed. They are supported on all IPDS

printers except the 4224, 4230, 4234, 4247, and 64xx. The use of font character sets provides greater consistency in the printed appearance of output from different printers.

Note: When a printer file is created and a character set and code page are specified for the font character set (FNTCHRSET) parameter, column spacing is done using this printer file level parameter. Any fonts or code pages specified in the DDS FNTCHRSET keyword are ignored and the font and code page specified in the printer file parameter FNTCHRSET are used.

Using the coded font (CDEFNT) parameter

This parameter is specified only for printer files with DEVTYPE (*AFPDS) when printing on IPDS-capable printers configured with AFP(*YES).

A coded font is the combination of a font character set and a code page. This combination is assigned a name and called a coded font.

Note: The coded font contains only the names of the font character set and code page. It does not contain the font and code page data.

A point size may be specified for outline fonts. It will be ignored for raster fonts.

You can find out which coded fonts come with the iSeries server by using the Work with Font Resources (WRKFNTRSC) command and specifying QFNTCPL for the library and *CDEFNT as the object attribute.

Note: If you have obtained coded fonts from other sources but have them in a different library, the WRKFNTRSC command can display the coded fonts located in that library.

You can specify a library for the coded font specified on the printer file. However, if the font character set and code page that make up the coded font are not in a library that is defined to your library list, the coded font is not found.

Special printer file considerations

Direct print considerations

When output data is written directly to the printer instead of output queue, the job is called a direct print job. When printing direct print job on an SCS printer, the file device type is changed to *SCS by OS/400. When printing on an IPDS printer that is configured with AFP(*NO), the file device type is changed to *IPDS by OS/400. If the print job has file device type *AFPDS, *USERASCII, *LINE, or *AFPDSLINE, the direct print job is not supported. See "Using a printer for both spooled files and direct print jobs" on page 141 for more information about direct printing.

Open considerations

The following considerations apply to opening printer files for spooled output:
- The output queue should be created for the type of output your program produces so that system operator intervention can be kept to a minimum while a printer writer is producing output. You should consider the following when creating an output queue:
  - What form of output is being produced (printer or diskette).
  - What kind of forms the output is printed on.

- What kind of protection you want to place on your data. (Do you want someone else to be able to display your data?)
- How many job separators you want.
- The SCHEDULE parameter specifies when output is to be made available to a printer writer. When a printer writer processes a specific file, it is dependent on things such as:
  - When the writer is started
  - Other output files on the queue
  - If the writer or the output queue is being held
- The parameters specified to produce output are saved until they are used by the writer.

Output considerations

The following considerations apply to output operations performed on spooled files:

Force-end-of-data considerations

The force-end-of-data (FEOD) operation can make part of the spooled file available to the writer unless SCHEDULE(*JOBEND) or HOLD(*YES) are specified for the file. This operation lets you write parts of a spooled file; for example, you can write one order at a time. You should not use the force-end-of-data operation for normal output. A new spooled file is started after each FEOD operation.

Close considerations

When the schedule value is *FILEEND, the output file is made available to the printer writer. The file resources used by the program are deallocated.

If an application program is writing data when the system ends abnormally, the spooled output is shown containing 0 pages on spool displays such as the WRKOUTQ, WRKSPLFA, and WRKJOB displays. Records that were stored in internal system buffers are lost.

Spooled files that contain no records (open and close, but no output) are automatically deleted by the system when the application closes the device file. Writers will not select these files for printing unless SCHEDULE(*IMMED) is specified and the writer selects the file before it is closed.

First-character forms-control data

For program-described printer files, you can also specify the print control information in the data itself. You can do this by including an American National Standard first-character forms-control code in position 1 of each data record in the printer file. (You cannot use first-character forms control and DDS on the same file.)

For information about the machine data, see Chapter 8, "Working with line data" on page 215.

To include the print control information in the data, you specify one of the following American National Standard first-character forms-control codes in the first position of each data record:

Control Code
     Action before Printing a Line

`' '`	Space one line (blank code)
`0`	Space two lines
`-`	Space three lines
`+`	Suppress space
`1`	Skip to channel 1
`2`	Skip to channel 2
`3`	Skip to channel 3
`4`	Skip to channel 4
`5`	Skip to channel 5
`6`	Skip to channel 6
`7`	Skip to channel 7
`8`	Skip to channel 8
`9`	Skip to channel 9
`A`	Skip to channel 10
`B`	Skip to channel 11
`C`	Skip to channel 12

Any other character in position 1 of a record defaults to a blank (the American National Standard code for spacing one line). If this occurs, the notify message CPF4916 is sent to the high-level language program once per file.

When you use first character forms control data for a printer file, the print control information created by the high-level language compiler is ignored. The character in position 1 of the record is used as the print control character for that record.

To create a program-described printer file that uses first-character forms-control data, specify the CTLCHAR parameter and, optionally, the CHLVAL parameter on the Create Printer File (CRTPRTF) command. CTLCHAR(*FCFC) specifies that the first character in every record is an American National Standard forms-control code.

The CHLVAL parameter allows you to associate a specific skip-to line number with an American National Standard channel identifier. For example, if you specify CHLVAL(2 20), channel identifier 2 is allocated with line number 20; therefore, if you place the forms-control 2 in the first position of a record, the printer skips to line 20 before printing the line.

Note: If the printer stops at a particular line number and the next record processed has a channel value forms-control number that is the same value as the line number the printer is on, the printer advances to that value (line number) on the next page. In the example in the paragraph above, if the printer had been on line 20 already, the printer would then have advanced to line 20 of the next page.

There is one exception to this method of printer advancement.

If the printer is positioned at the top of the page (line 1) and the channel value forms-control value is line 1, the printer does not advance to a new page.

Each control identifier can be specified only once on the parameter. If no line number is defined for a channel identifier and that channel identifier is encountered in the data, the printer takes the default of spacing one line before printing.

In the following example, a file, PRTFCFC, that uses first-character forms-control data is created:

```
CRTPRTF  FILE(QGPL/PRTFCFC) OUTQ(PRINT) CTLCHAR(*FCFC) +
CHLVAL((1 1) (2 10) (12 60))
```

The printer output is spooled to the output queue PRINT. Channel identifier 1 is associated with line 1, channel identifier 2 is associated with line 10, and channel identifier 12 is associated with line 60.

Printer font support

Characters per inch versus font

Most SCS printers use the CPI parameter on the CRTPRTF, CHGPRTF, or OVRPRTF command to determine the pitch (characters per inch) of the printed output. However, the 3812, 3816, 5219, and all IPDS printers use the FONT parameter to select both an implied pitch (characters per inch) and a font style. For these printers, the CPI parameter is ignored except when converting the page size (PAGESIZE) from *UOM to *ROWCOL. Conversely, the FONT parameter is ignored on printers where it is not applicable.

To provide flexibility in your print job, the pitch of the font identifier (FONT parameter) should match, where possible, the value specified on the CPI parameter. By setting the FONT parameter to FONT(*CPI), the system selects a font of the same pitch as the CPI parameter value. A print job intended for a printer that supports fonts can then be printed on another printer without significant change in the appearance of the printed output. For example, a printer file which has FONT(222), Gothic font with 15 pitch, and CPI(15) could print on a 3812, 4224, or 5219 Printer (which use the FONT parameter) or also could print on a 4214, 4230, 4234, 4247, 5224, or 5225 Printer (which support 15 characters per inch). If this print job were directed to a printer that only supports 10 characters per inch, then printer file redirection would be used.

For more information about redirecting spooled files to SCS printers, go to "Spooled file redirected to SCS printers" on page 112.

When using an SCS externally described printer file, normally the value specified in the CPI parameter is used to position fields on the printed page. For example, if a printer file has 10 characters per inch specified, and FIELDA is specified to start in column 51, then there would be 50 blanks to the left of FIELDA (50 blanks at 10 characters per inch is 5 inches).

RSLH114-0

For the IPDS externally described printer files, the pitch implied by the FONT parameter on the CRTPRTF, CHGPRTF, and OVRPRTF commands is used instead of the CPI parameter or the value of the FONT DDS keyword to determine the starting column of fields on a printed page. In the example above, if CPI(10) and

FONT(087) had been specified, and the printer specified was one that supports fonts, then FIELDA specified to start in column 51 would be 50 blanks at 12 characters per inch (implied CPI value for font 087) or 50/12 inches, which is 4.167 inches in from the left margin of the paper. The default for FONT is (*CPI). When *CPI is selected, the OS/400 program automatically selects a font of the pitch specified on the CPI parameter.

Proportionally spaced and typographic fonts

All IPDS printers support proportionally spaced fonts. For proportionally spaced fonts, characters vary in width depending on the character being printed (for example, **i** is a narrow character, and **W** is a wide character). See Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377 for a list of all supported fonts. The implied characters-per-inch column in this table lists the value of the width of a blank character for the font selected.

When using proportionally spaced fonts, fold and truncation (FOLD parameter) may not work as intended. This is because the system does not keep track of the width of each individual character.

The following printers also support typographic fonts: the 3812, 3130, 3160, 3816, 3820, 3825, 3827, 3829, 3831, 3835, 3900, 3916, 3930, 3935, 4028, 4312, 4317, 4324, InfoPrint 3000, and the InfoPrint 4000. You can specify a typographic font by indicating point size (height of font). A point measures 1/72 of an inch. An 8-point font would be 1/9 of an inch high, and a 24-point font would be 1/3 of an inch high. When using fonts that contain tall characters, it may be necessary to double or triple space, to avoid having lines overlap when printing the page.

Because proportionally spaced and typographic fonts have characters of variable widths, care should be taken with the use of underlining and overstriking. The highlighting or underlining method of printing a line with a space after (SPACEA) value of 0 followed by printing another line may not work correctly.

Proportionally spaced and typographic fonts may be specified on the CRTPRTF, CHGPRTF, or OVRPRTF command when using an externally described printer file. The point size may be specified for typographic fonts. The point size is ignored for fonts that are not typographic. As noted above, the implied characters-per-inch value of the font identifier specified on the FONT parameter is used to position fields on a printed page. The same rule for positioning fields on a printed page is used with proportionally spaced and typographic fonts. The width of a blank character is used to position fields on a page. Output should be tested to see that using externally described printer files with proportionally spaced fonts produces satisfactory results because overprinting and gaps can occur in the output. Because of slight adjustments made for position checks, it is recommended to not print on line 1 when specifying 8 or 9 LPI on an IPDS printer.

Note: The amount of printed space for a field varies depending on which characters are in a field. Enough space should be left between fields to allow for the widest characters (uppercase characters) expected in that field.

In the previous example, if CPI(10) and FONT(1351) had been specified, and the printer used was a 3812 Printer, then FIELDA specified to start in column 51 would be 50 blanks at 17.14 characters per inch (implied CPI value for font 1351) or 50/17.14 inches, which is 2.975 inches from the left margin of the paper.

In this example, FIELDA would start 2.975 inches from the left margin of the paper regardless of how many fields were defined to its left. When using an increment value (+n) instead of a column number (positions 42 through 44 in DDS specification), fields are positioned the same for proportionally spaced fonts as they are for fixed pitch fonts. That is, the field is positioned based on the width of the blank for the specified font. The following example illustrates that using either absolute column numbers or relative increment numbers (+n) will supply the same result.

```
A *
A * USE ABSOLUTE COLUMN NUMBER TO POSITION FIELDS
A *
A          R PRTOUT                    SKIPB(1)
A            FIELDA      10         11 SPACEA(1)
A            FIELDB      15         31 SPACEA(1)
A            FIELDC      10         51 SPACEA(1)
A
A *
A * USE INCREMENT (+n) TO POSITION FIELDC2
A *
A          R PRTOUT2                   SKIPB(1)
A            FIELDA2     10         11 SPACEA(1)
A            FIELDB2     15         31 SPACEA(1)
A            FIELDC2     10         +5 SPACEA(1)
A
```

RSLH115-3

Font substitution

If the font identifier specified on the FONT parameter is not supported by the printer being used, printer data management selects a substitute font that is supported by the printer (if possible). A substitute font is at the same pitch or a higher pitch whenever possible to ensure that as much data as possible fits on the printed page. A complete list of fonts supported and the substitute font selected for each printer that supports the FONT parameter is shown in Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377.

If a substitute font cannot be used, spool redirection is done. An inquiry message is sent to the message queue associated with the device or printer writer. The inquiry message gives you the option of holding or printing the file. If the print option is used, then the spooled output file is reformatted with the print attributes of file QPSPLPRT. Output may not look as intended. Refer to the printer's reference manual for information on what conditions cause the substitute font to not be used.

For the SCS 3812, 3816 and 5219 Printers, font substitution can be made only at the file level. For font changes made in the document, font substitution is not done. In this case, spool redirection (described previously) is used.

IPDS printer FONT parameter considerations

When a printer has AFP(*YES) specified in the printer device description and the DEVTYPE specified for the printer file being used is *AFPDS, the FNTCHRSET, CDEPGE, and CDEFNT printer file parameters may be used to select a font resource to be downloaded to the printer. This applies to all IPDS printers except the 4224, 4230, 4234, 4247, and 64xx. If these parameters are not used, then the value specified in the FONT printer file parameter is used.

When FONT(*DEVD) is specified on the CRTPRTF, CHGPRTF, and OVRPRTF commands, the following limitations are imposed:

- Bar codes are positioned on the page assuming a 10-pitch font is specified in the device description.
- When using a combination of bar code (BARCODE), page rotate from a printer file or DDS (PAGRTT), and character size (CHRSIZ) parameters in an externally described printer file, unpredictable results may occur. This is because the printer device FONT parameter value is not known when the spooled file is created.
- The data stream created may be longer than if a specific font were selected. This means the spooled file may take more storage in the output queue. Fields are positioned with spaces (hex 40) between them instead of using commands to specify the location where a field is to be placed.
- If a proportionally spaced font is used with a field in the file, any following fields may not be positioned in the column used if a specific font were specified at the file level. This is caused by the variable width of the characters in the proportionally spaced font, which are followed by spaces (hex 40) to position the next field.
- If the value of the FONT parameter on the printer device description is *DEVD or 0, font 011 is selected.

The maximum number of fonts that can be sent to a printer file is 48. When more than 48 fonts are requested, an error message is sent.

A slight adjustment is sometimes made to the first or last line of a page when the lines per inch (LPI) parameter value is greater than 6. This adjustment prevents IPDS printers from reporting position check errors due to part of a character printing off the top or bottom of the page. For the first line on a page, a slight downward adjustment is made. For the last line on a page, a slight upward adjustment is made. This adjustment is about 1/72 of an inch. No other lines on the page are adjusted. This adjustment is made only for spooled files with DEVTYPE of *SCS or *IPDS when printed on IPDS-capable printers. It is recommended that the first line on a page not be used for printing if the lines per inch (LPI) parameter on the printer file is 8 or greater.

Note: If an optical character recognition (OCR) font is specified with a non-OCR code page, the code page is changed to an OCR code page. If a non-OCR font is specified with an OCR code page, the font is changed to an OCR font.

Using graphic symbol sets

IPDS printers allow the selection of graphic symbol sets as fonts when using the DDS font keyword.

To find which graphic symbol sets are available on your system, enter the CL command:

```
DSPOBJD OBJTYPE(*GSS) OBJ(QGDDM/*ALL)
```

Replacing on unprintable characters

You can replace unprintable characters in your data before the data is written to the printer or to a spooled file by specifying RPLUNPRT(*YES) on the CRTPRTF, CHGPRTF, or OVRPRTF commands. The replacement of an unprintable character depends on the printer being used and the hexadecimal value of the unprintable character.

The RPLUNPRT value must be selected before the spooled file is created. Once a spooled file is in an output queue, changing the RPLUNPRT value has no effect on that particular spooled file.

- When RPLUNPRT(*YES) is specified, any characters in the range hex 00 through hex 3F, and hex FF, are replaced. The default replacement character is a blank. The character which a printer cannot print varies depending on the type of printer.
- When RPLUNPRT(*NO) is specified, no translation of the data stream is made. Any characters in the range hex 00 through hex 3F, or hex FF, may cause undesirable results. These characters are in the range used by printer control characters.

  For most characters in this range, the printer signals an unrecoverable error and the spooled file either is held in the output queue or is not processed. Some characters in this range control forms action and character representation on the printer and, as a result, additional skipping or spacing may occur. If control characters are placed in the data, system functions such as displaying or copying spooled output files and restarting or backing up a printer may produce results that cannot be predicted.

  If the hexadecimal value of the unprintable character is hex 40 through hex FE, a message is sent to the message queue associated with the printer. The message gives you the option to end the writer, hold the spooled file, ignore the error and continue printing, or select a page number where printing should be restarted. If the ignore option is taken, then unprintable characters continue to be reported. If the option to start again (specify page number) is taken, all unprintable characters are replaced with blanks and you receive no more notification for unprintable characters.

Considerations for the 4245, 5262, 6252, and 6262 printers

The printer translates lowercase characters to uppercase characters when using a print band that does not contain lowercase characters. If your print job contains other characters that are not on the print band, they can be translated to blanks by specifying RPLUNPRT(*YES) for the printer file.

CL commands that you can use to do this are: CRTPRTF, CHGPRTF, and OVRPRTF.

A print band is selected by switches on the 5262 operator's panel. The operator must select both a language ID and a band image using these switches.

The 4245, 6252, and 6262 Printers detect the print band the printer is using.

When a print band changes for a print job, no inquiry message is sent to the message queue associated with the printer writer. You can specify a different form type for that job, such as blank payroll forms or blank invoices. A message to change the form type is sent to the printer writer message queue to notify the operator of the change to the print band.

Using alternative character sets and code pages for printer output

Character sets are used with code pages to determine how each character will appear in the printed output. Code pages consist of hexadecimal identifiers (code points) assigned to character identifiers. For example, in code page 037 (EBCDIC), the letter e is assigned a code point of hex 85.

In multinational environments, data in one national graphics character set may need to be printed on devices that support another national character set. This is particularly true of characters with accents and other characters with diacritical marks (such as ç, ñ, and ü). In this section, these characters are called **extended alphabetics**.

For example, assume that a physical file on the system contains data in the Basic French character set, and includes the character é. In the code page used with the Basic French character set, this character is hex C0. The data could have been entered on a display device that can handle the character or could have been sent to the system from another system over a communications line. When hex C0 is sent to a printer that is set up for the United States Basic character set, the hex C0 is printed as {. Depending on the printer and the hexadecimal value sent, the hexadecimal value could be an unprintable character. The way the printer handles a specific hex code point (for example, hex C0) depends on the current value of the CHRID parameter in the printer file. You can specify the following parameter values for the CHRID parameter:

- With an explicit value specified for the CHRID parameter, the printer interprets the data as if the data were in the character set and code page specified.
- With CHRID(*SYSVAL) specified, the printer file takes the value specified in the QCHRID system value when the output is created.
- With CHRID(*DEVD) specified, the printer uses the CHRID that was set with the device control panel or that was specified when the printer device description was created.
- If you have specified CHRID(*JOBCCSID) the printer interprets the data as if it were in the character set and code page that are associated with the CCSID for the current job. For more information, see theGlobalization topic in the iSeries Information Center.
- With CHRID(*CHRIDCTL) specified, the printer file checks the CHRIDCTL job definition attribute to determine whether to use *JOBCCSID or *DEVD on the CHRID command parameter for the job.

Not all printers can handle all CHRID parameter values. If a CHRID is specified for a printer on which that CHRID is not supported, a message is sent to the operator. Go to Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377 for a description of which printers support which extensions.

For program-described printer files, the value of the CHRID parameter determines the code page and character set used to print the data. However, for externally described printer files, the CHRID parameter is used only for fields that also have the CHRID DDS keyword specified. Fields that do not have the CHRID DDS keyword use the code page and character as if CHRID(*DEVD) had been specified for the CHRID parameter on the printer file.

For printers with AFP(*YES) in the device description, the printer file CDEPAG and FNTCHRSET parameters can be used to select a code page when the printer file DEVTYPE parameter is *AFPDS. This applies to all IPDS printers except 4224, 4230, 4234, 4247, and 64xx.

The following shows how extended alphabetics are handled in printer output:

```
┌──────────┐        ┌──────────┐        ┌──────────┐        ┌──────────┐
│ Physical │  Read  │Application│ Write │ Spooled  │        │ Printed  │
│ File     │───────▶│ Program  │──────▶│ File     │───────▶│ as       │
│          │        │          │        │          │        │          │
│ ┌──────┐ │        │ ┌──────┐ │        │ ┌──────┐ │        │ ┌──────┐ │
│ │ Renée│ │        │ │ Renée│ │        │ │ Renée│ │        │ │ Renée│ │
│ └──────┘ │        │ └──────┘ │        │ └──────┘ │        │ └──────┘ │
└──────────┘        └──────────┘        └──────────┘        └──────────┘
                                                          RS3H020-0
```
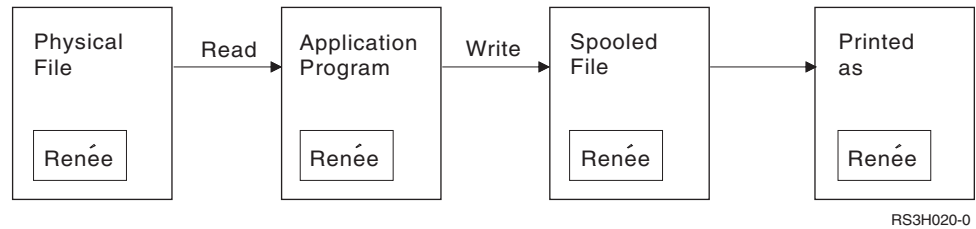
Assume that a record in a physical file contains a field with the value Renée. An
application program reads the record from the physical file, and writes a record
containing the data to the spooled file. The output field in the printer file that
describes how Renée is to be printed has the CHRID DDS keyword specified,
indicating that the printer is to interpret extended alphabetics. (The graphic
character set 288 and code page 297 are specified for the interpretation in either the
printer file or the QCHRID system value. Code page 297 is used for French
language.)

When printing the data, the printer interprets hex C0 as specified in character set
288 and code page 297. If character set 101 and code page 037 had been selected,
hex C0 (é) would have been printed as {.

One of the following CHRID values (graphic character set and code page) must be
specified to print fonts OCR-A and OCR-B on the IPDS printers:

    580 340

    590 340

    697 892

    697 893

Print text

You can specify on the printer file a line of text that is to be printed at the bottom
of every page. This line of text is called the print text and is set by using the
PRTTXT parameter on the CRTPRTF, CHGPRTF, or OVRPRTF commands. Up to 30
characters are allowed in the line of print text. The 30 characters are centered at the
bottom of the page, 2 lines below the overflow line. If the user already has data to
print on the line that the print text goes on, the print text is bumped down to the
next blank line on the page. If no lines are blank, the print text is printed on the
last line of the page.

Notes:

1. For externally described printer files with DEVTYPE(*AFPDS) using the DDS
   POSITION keyword, the print text is positioned by ignoring the location of any
   data placed on the page by records using the POSITION keyword. If all the
   data on the page is positioned using the DDS POSITION keyword, the print
   text is located on the overflow line.

2. If a host resident font is specified on the printer file, 10 characters per inch is
   used to calculate the location of the text specified on the PRTTXT parameter.

A system value, QPRTTXT, can be used to specify the print text so that the same
text can appear on all files printed on the system. Also, the print text can be taken
from the job description, so that all files created from a particular job can have the
same print text.

Print text is useful for printing a security classification on each page. It can also be
used to print a company name or slogan on each page.

Editing output fields

The system provides editing support that makes fields more readable when they are printed. With the system editing support, you can do the following:

- Suppress leading zeros
- Punctuate a field with commas and periods to show decimal column and to group digits by threes
- Print negative values with a minus sign or CR to the right
- Print zero values as zeros or blanks
- Print asterisks to the left of significant digits to provide asterisk protection
- Print a currency symbol corresponding to the system value QCURSYM

The system provides this editing support with edit codes and edit words. Edit codes are a defined set of editing patterns. You identify these by name, and the system edits a field according to the pattern defined by the named edit code. Edit words are edit patterns that you define to produce the desired results. Edit codes cover most commonly used editing requirements. You need to use the edit word support only for those editing needs not covered by edit codes.

There are two methods of using edit codes and edit words. Which one you use depends on how you define the printer file and how it is used in an application program. If your application is using program-described data, your high-level language may allow you to identify edit codes or create your own edit words. If your application is using externally described data, the edit code (EDTCDE) DDS keyword allows you to identify an edit code; the edit word (EDTWRD) DDS keyword allows you to define your own editing pattern.

The system provides several edit codes:

- 1 through 4
- A through D
- J through M
- X through Z

The editing patterns defined by these codes are described in theDDS Reference topic in the iSeries Information Center.

User-defined edit codes

You can also define five edit codes to provide more editing function than is available with the OS/400 edit codes, and to handle common editing functions that would otherwise require the use of an edit word. These are called user-defined edit codes. For example, you may need to edit numbers that include hyphens (like some telephone numbers), or more than one decimal point. You can use user-defined edit codes for these functions. These edit codes are named QEDIT5, QEDIT6, QEDIT7, QEDIT8, and QEDIT9 and can be referred to in DDS or a high-level language program by number (5, 6, 7, 8, or 9).

These edit codes are created by using the Create Edit Description (CRTEDTD) command. Edit descriptions are always placed in library QSYS. They cannot be moved or renamed; only one occurrence of each is allowed. Edit descriptions have an object type of *EDTD.

IBM supplies a version of each of the QEDIT edit codes. You can use these edit descriptions as they are, or you can delete them and create your own. See the DDS Reference topic in the online Information Center for more information about using these edit descriptions.

Before using any of the user-defined edit codes, you should check its contents on your system, since it may have been changed from the IBM-supplied version. The Display Edit Description (DSPEDTD) command can be used to display the contents of a user-defined edit code.

Changing a user-defined edit code description does not affect any application or printer file that has already been created using that edit description. If you want your application to use the changed edit description, you must either create the high-level language program again (if the edit code is used in the program) or create the file again (if the application is using an externally described file that contains EDTCDE keywords).

Effect of changing fields in a file description

When a program using externally described printer files is compiled, the compiler extracts the file descriptions for the files referred to in the program and makes these file descriptions part of the compiled program. When you run the program, you can verify that the record formats with which the program was compiled are the current record formats. To do this, you use the LVLCHK parameter on the create file command when the file is created.

The system assigns a unique level identifier for each record format when the file it is associated with is created. The system uses the information in the record format description to determine the level identifier. This information includes the name of the record format, the names, attributes, and order of the fields in the format, the indicators used, and the names and the order of the indicators in the record format. If you use the INDARA keyword to remove the indicator from the output buffer, the indicators used are not included in the level identifier information.

When the file is opened, if level checking is specified (LVLCHK parameter), the system does a format-by-format comparison of the level-checking values specified in the program to the level-checking values specified in the printer file. If any of the formats specified in the program do not exist in the file, or if any of the level checking values are different, an error occurs. Formats can be added to or removed from a printer file without affecting existing application programs that do not use the added or deleted formats.

You should display the file description to determine if the changes affect your program. You can use the Display File Field Description (DSPFFD) command to display the file description or, if you have the source entry utility (SEU), you can display the source file. Not every change in a file necessarily affects your program. You may not have to recompile your program. If you do not have to recompile your program, you should specify LVLCHK(*NO) for the file (CHGPRTF or OVRPRTF command).

You can add a field to the end of a printer file record format without having to recompile your program as long as you do not want to use the field in your program. If you delete a field from the end of the record format, you do not have to recompile your program if you are not using the field. However, if you add a field to or delete a field from a record format anywhere other than at the end, you

must recompile your program. Otherwise, the field offsets in the record passed to and from the program are wrong for processing.

In general, anything that changes the length or position of any fields in the record format used by the program will require that the program be recompiled.

Redirecting output

Spooled or nonspooled output intended for a printer can be redirected to another printer. However, each file is checked to ensure that the file attributes (device type, number of lines per inch, number of characters per inch, page length, and page width) and any advanced functions used by the file (such as variable LPI, variable font, or defined characters) are valid on the new printer.

Nonspooled output

When a nonspooled file is redirected, and the printer file attributes do not match the new printer, one of the following occurs:

- If the printer file specifies a characters-per-inch value not supported by the device being used, a diagnostic message (CPF4057) is sent to the program message queue and the data is printed at 10 characters per inch. If the page width is greater than 132 characters, the records are folded.

   Folding is not supported on IPDS printers.:
   1. If the printer file specifies a lines-per-inch value not supported by the device being used, a diagnostic message (CPF4056) is sent to the program message queue, and the data is printed at 8 lines per inch.
   2. If the page length is greater than the maximum length allowed for the printer being used, the printing ends with an escape message (CPF4138).
   3. If the printer file specifies special device requirements (such as use of certain DDS keywords) that are not supported by the device being used, then a diagnostic message is sent to the program message queue and the special function is ignored.

Spooled files

When a spooled file is redirected to another printer, the spooled file cannot be printed without change if any of the spooled output file attributes are not supported by the printer device. For example, some printer device attributes that might not be supported are:

- Page size
- Output drawers
- Print quality
- Lines per inch
- Characters per inch

Spooled file redirected to SCS printers

The following section describes the actions taken when a spooled file is redirected to an SCS printer and cannot be printed without change (SCS printers include the 3812, 3816, 4214, 4234, 4245, 4247, 5219, 5224, 5225, 5256, 5262, 6252, and 6262 Printers):

- An inquiry message is sent to the message queue of the writer if any of the following is true:
   - The spooled file uses the IPDS data stream (DEVTYPE(*IPDS))
   - The spooled file attributes are not supported by the printer

– The special device requirements used by the spooled file are not supported by the printer

The inquiry message allows these options:
– End the writer
– Print the spooled file with lines folded when the lines are longer than the width of IBM-supplied printer file QPSPLPRT
– Print the spooled file with lines truncated when the lines are longer than the width of IBM-supplied printer file QPSPLPRT
– Hold the spooled file and process the next file on the output queue

If the spooled file is printed, results may be unpredictable because the file is printed using the printer attributes specified in the IBM-supplied printer file QPSPLPRT, and all advanced functions used by the spooled file are removed. Functions removed include:

DDS keywords:

CHRID
Graphic character set and code page

CHRSIZ
Character size (width and height)

CPI   Characters per inch

DFNCHR
Define character

DRAWER
Paper drawer selection

FONT Font selection

LPI   Lines per inch

PAGRTT
Page rotation

PRTQLTY
Print quality

TRNSPY
Transparency

Other print functions:

Drawer change in document
Font change in document
Lines-per-inch change in document
Page rotation in document
Subscript and superscript

- If the spooled file specifies a characters-per-inch value not supported by the printer, an inquiry message is sent to the message queue of the writer with the option to:
  – End the writer
  – Print the spooled file at 10 characters per inch with lines folded when the lines are longer than the width of IBM-supplied printer file QPSPLPRT
  – Hold the spooled file and process the next file on the output queue

- The 5219 Printer is an exception to the above cases if the only mismatch between the spooled file and printer is the HIGHLIGHT special device requirement that the file contains. When this occurs, an inquiry message is sent to the message queue of the writer with the option to:
  - End the writer.
  - Print the spooled file without highlighting but keep all other advanced functions used by the file.
  - Attempt to print the spooled file without changing. (If this is not successful, the file will be held on the output queue.)
  - Hold the spooled file and process the next file on the output queue.

  If the spooled file is printed, the resulting output closely resembles how the file was intended to look. This is because the attributes specified by the spooled file were used and advanced functions were kept.
- Documents created on other systems may contain print controls that are not supported by the 5219 or 3812 Printers. These controls may include variable form size, output drawer, print quality, lines per inch, characters per inch, character identifier, or justification. If this occurs, an inquiry message is sent to the message queue of the writer with the options to:
  - End the writer.
  - Print the spooled file with unsupported values changed to values which are supported by the printer.
  - Attempt to print the spooled file without changing. (If this is not successful, the file will be held on the output queue).
  - Hold the spooled file and process the next file on the output queue.

If the spooled file is printed, the file attributes from the spooled file are used and all advanced functions that are valid for the device are kept. The resulting output should closely resemble what the file was intended to look like, though it still may not print exactly as intended because of the unsupported values.

Spooled files redirected to IPDS printers

IPDS printers include: the 3130, 3160, 3812, 3816, 3820, 3825, 3827, 3829, 3831, 3835, 3900, 3912, 3916, 3930, 3935, 4028, 4224, 4230, 4234, 4247, 4312, 4317, and 4324. InfoPrint 20, InfoPrint 32, InfoPrint 3000, and InfoPrint 4000 are also IPDS printers. The following describes the actions that are taken when you spool a file to an IPDS printer:

- The message queue will receive an inquiry if the spooled file uses the SCS data stream (DEVTYPE(*SCS)) and contains DBCS (double-byte character set) data. It will also receive an inquiry if it has a page length greater than that supported by the printer (for both SNA character string (SCS) and IPDS files). You can choose any of the following options:
  - End the writer
  - The spooled file will print with lines that are truncated when the lines are longer than the width of IBM-supplied printer file QPSPLPRT
  - Hold the spooled file and process the next file on the output queue

Printing results may be unpredictable, because the printer will use the printer attributes that are specified in QSPLPR, the IBM-supplied printer file. All advanced functions that are specified in the spooled file will be removed. Functions removed include:

DDS keywords:

CHRSIZ
> Character size (width and height)

CPI Characters per inch

DFNCHR
> Define character

DRAWER
> Paper drawer selection

PAGRTT
> Page rotation

TRNSPY
> Transparency

Other print functions:

> Drawer change in document
> Font change in document
> Lines-per-inch change in document
> Page rotation in document
> Subscript and superscript

- The message queue will receive an inquiry message for the following: if the spooled file uses the SCS data stream, (DEVTYPE(*SCS)), does not contain DBCS data, and uses special device requirements. Special device requirements include graphics, defined characters, transparencies, variable font, and enhanced 3812 fonts. It will also receive an inquiry if it uses a proportionally spaced font for the FONT parameter of the file. You can select any of the following options:
  - End the writer
  - Transform the spooled file to IPDS format and print
  - Hold the spooled file and process the next file on the output queue

  If the file prints, its existing attributes will be used. Advanced functions remain intact, except graphics, defined characters, justification, and transparencies. The transformation to IPDS format should substantially keep the integrity of the text data. However, the file may still not print exactly as intended. Unsupported font pitches, font spacings, and character identifiers are changed to the closest approximation valid on the printer.

- The writer automatically transforms the file to an IPDS file and prints it if the following is true: the spooled file uses DEVTYPE(*SCS), does not contain DBCS data, does not use defined characters, graphics, transparency, variable fonts, or enhanced 3812 fonts. The printer file also can not use a proportionally spaced font for the FONT parameter of the printer file. The spooled file may not print exactly as intended. For unsupported font pitches, font spacings, and character identifiers, the printer writer uses the closest approximation available on the printer.

- The message queue of the writer receives an inquiry message if the spooled file uses the IPDS data stream (DEVTYPE(*IPDS)), but uses advanced functions not supported by the printer. Options made available are:
  - End the writer
  - Print the spooled file, but dropping unsupported advanced functions from the file
  - Hold the spooled file and process the next file on the output queue

3812 and 3816 SCS printer considerations

When automatic configuration is run for 3812 and 3816 printers, the printers report as a 5219 Printer. The first time the printer is used, the iSeries server sends some commands to the printer that allow the system to distinguish between a 5219 Printer and a 3812 or 3816 SCS printer.

However, this happens after the open processing has been done for the first output to print. The first output to print can be direct output or a spooled file. This means, for the first output printed, the system treats the printer as a 5219 Printer. For example, this means there is no page rotation for that first printed output.

In order for the system to recognize a 3812 or 3816 SCS printer, the printer writer must complete processing. After the first output has been printed, and a new printer writer is started, the system recognizes the printer as a 3812 or 3816 SCS printer.

Once the system recognizes a printer as a 3812 or 3816 SCS, it remembers the true printer type until the printer device description is deleted.

3835 printer considerations

The 3835 Model 1 printer has a no-print border. In this area, about 1/6 inch from all edges of the page, data will not print.

The iSeries server adjusts the positioning of the printed text on the page to compensate for the no-print border. For example, if your application program prints text in the top 1/6 inch or left 1/6 inch, all the text will print. The iSeries server adjusts the starting printing position to 1/6 inch from the top and left of the page causing all printed data on the page to be shifted to the right and down by 1/6 inch. If your application relies on printing data at a certain point on the paper, you may have to change your application to compensate for this adjustment or use the margin values of 0 on the printer file.

Note: The 3835 Model 2 does not have a no-print border. The iSeries server does not adjust the position of printed text for this printer. When directing output from printing on a 3835 Model 1 to a Model 2, the difference in how the no-print border is used must be considered.

3912, 3916, and 4028 printer considerations

The 3912, 3916, and 4028 printers have a no-print border. In this area, about 1/6 inch from all edges of the page, data will not print.

If the 3912, 3916, or 4028 is configured AFP(*NO), you may have to adjust your application programs. For example, if your application program prints text in the top 1/6 inch or left 1/6 inch, that text will not appear on the page.

If the 4028 is configured AFP(*YES), the iSeries server adjusts the positioning of the printed text on the page to compensate for the no-print border. For example, if your application program prints text in the top 1/6 inch or left 1/6 inch, all the text will print. The iSeries server adjusts the starting printing position to 1/6 inch from the top and left of the page causing all printed data on the page to be shifted to the right and down by 1/6 inch. If your application relies on printing data at a certain point on the paper, you may have to change your application to compensate for this adjustment.

To print as close to the edge of the page as possible:
- Use PAGRTT (*COR)
- Use MULTIUP(2) or (4)
- Configure the printer with AFP(*YES)

This positions the starting origin of the page to the edge of the printable area.

Printing a graphic along with other output

The command word #$@INCLGRPH allows a user to include a graph anywhere in the data printed by high-level languages.

To include graphics with other program output, a special control record is used. The format for this control record is:

`#$@INCLGRPH filename,x,y,w,l`

Notes:

1. There must be only one space between the command word and the parameters.
2. All five parameters must be specified. A default value (the default values are included in the list below) for parameters x, y, w, and l can be obtained by omitting any value. For example, a valid control record with defaults for parameters x, y, and l would look like:

   `#$@INCLGRPH filename,,,9.5,`
3. The #$@INCLGRPH control record should be in a print record by itself, because any other data along with it may be considered as parameters.
4. Parameters should immediately follow one another, separated by commas, using no blanks.
5. The characters INCLGRPH must be all uppercase.
6. The #$@INCLGRPH control record must begin in the first column.
7. The #$@INCLGRPH control word is used with characters from code page 500. For example, in code page 500, @ is hex '7B', $ is hex '5B', and @ is hex '7C'. Other code pages may use other characters in the#$@INCLGRPH control record. You will need to change the characters depending on the code page being used.

The parameters are defined below. Parameters x, y, w, and l define the area on the page where the graphics file will be printed. Parameters x and y define the upper left corner of the graphics area, and parameters w and l define the size of the graphics area.

filename
> The name of the graph object file to be included. If the file has more than one member, the last member is used. The library containing the file must be in your library list.

x
> The distance, in inches, from the left edge of the page to the left edge of the graphics area on the page. The default is 0.

y
> The distance, in inches, from the top of the page to the top edge of the graphics area on the page. The default is 0.

w
> The width of the graphics area, in inches. The default is the width of the current page you are using.

l
> The length of the graphics area, in inches. The default is the length of the current page you are using.

Parameters x, y, w, and l can be specified in decimal form in any combination of xx.xx, where x is any number from 0 through 9. The specified value cannot be more than 45.50 and, if a 0 value is specified, the result is the default value for that parameter.

If there are any errors found in the control record or an error occurs while processing the graphics file, the control record is printed as normal text data. The graphics file to be used must be in a format acceptable to the printer. For IPDS devices, this format is level DR/2 of the Graphic Object Content Architecture (GOCA). See the manual *Graphic Object Content Architecture* , SC31-6804, for more information about GOCA.

Special printer file considerations for AFPDS

If the device type (DEVTYPE) parameter for the printer file is *AFPDS, certain considerations apply to some printer file parameters and to the sending of spooled files to other systems.

Considerations for printer file parameters

- Overflow (OVRFLW) parameter

  Overflow is not signalled for externally described printer files (DDS) for record formats that use absolute positioning. In addition, overflow is determined by using the margin offset down value. For example, if the margin offset down value is .5 inches, the overflow line is line 60, and the lines per inch value is 6, overflow is signalled when line 60 is printed on the page. This is 10.5 inches down the page.

- Character Identifier (CHRID) parameter

  The CHRID parameter of the printer file is ignored if a font character set (FNTCHRSET) or coded font (CDEFNT) parameter is specified. One exception to this is if a file contains UCS-2 data that will be converted to EBCDIC data. In this case, the CHRID parameter determines the target conversion CCSID.

- Page Rotation (PAGRTT) parameter

  Overlays and page segments are not automatically rotated based on the PAGRTT parameter of the printer file.

- Page Size (PAGESIZE) parameter

  If the unit of measure is *ROWCOL, and either a coded font or font character set is specified on the printer file, the page width is calculated using 10 characters-per-inch.

Considerations for sending an AFPDS spooled file to another system

Because the actual printer device is not known when a spooled file is created, some parameters specified on the printer file cause a default setting to be used in the created spooled file. This is done to provide values in the data stream of the spooled file in the event it is sent to another system.

On the system the spooled file was created on, the correct values (as determined for the printer you want the spooled file printed on) are substituted before the spooled file is printed.

The following parameters cause a default setting to be used:

- If CHRID(*DEVD) or an externally described printer file is used, the CHRID system value (QCHRID) is substituted.
- If FONT(*DEVD) is used, font 11 is substituted.
- If FORMFEED(*DEVD) is used, drawer 1 is substituted.
- If PAGRTT(*DEVD), PAGRTT(*AUTO), or PAGRTT(*COR) is used, page rotation of 0 is substituted.

Note:

- On iSeries servers, offset stacking of printed output is used. As the job finishes, the paper tray moves, offsetting the stack of paper to make it easier to distinguish between finished jobs. Because of this, the data stream that is created on the iSeries server contains the control to indicate that offset stacking should be used. If the spooled file is sent to a system that does not support offset stacking, an error message may be issued.
- Use of the DDS DRAWER and PAGRTT keywords cause the OS/400 to generate an AFPDS datastream that is not completely supported by the AFP viewer and some of the PSF products available on other IBM platforms. If the spooled file needs to be viewed by the AFP viewer or printed on other IBM platforms, do not use the DRAWER and PAGRTT keywords in DDS. Use the INVMMAP keyword instead to change the drawer or page rotation within the spooled file.

Special DDS considerations for AFPDS

AFP and Print Services Facility for OS/400 (PSF/400): To use advanced function printing (AFP) support on the iSeries server to print on IPDS printers, you must install PSF/400.

Please see "Using Print Services Facility™ for OS/400® (PSF/400)" on page x for information on when PSF/400 is required. If you have additional questions about PSF/400, contact your IBM representative.

Following is a list of DDS keywords that are valid for printer files that have the printer device type (DEVTYPE) parameter value specified as *AFPDS. Restrictions on DDS keywords are contained in this list as well. For more detailed information about DDS keywords, see the DDS Reference: Printer Files topic in the iSeries Information Center.

ALIAS	GDF
BARCODE	HIGHLIGHT (Only applies to output printed using a printer resident font. If a coded font (CDEFNT) or a font character set and code page combination (FNTCHRSET) is specified, the HIGHLIGHT keyword is ignored and a message issued.)
BOX	IGCCDEFNT
CCSID	INDARA
CDEFNT	INDTXT
CHRID (Only applies to output printed using a printer resident font. If a coded font (CDEFNT) or a font character set and code page combination (FNTCHRSET) is specified, the CHRID keyword is ignored and a message issued.)	INVMMAP

CHRSIZ	LINE
COLOR (Color is ignored if your printer does not support color printing.)	MSGCON
CVTDTA	OVERLAY
DATE	OUTBIN
DATFMT	PAGNBR
DATSEP	PAGRTT
DFT	PAGSEG
DLTEDT	POSITION
DOCIDXTAG	PRTQLTY
DRAWER	REF
DTASTMCMD	REFFLD
DUPLEX	SKIPA (Not allowed at the file level in a spooled file with printer device type *AFPDS.)
EDTCDE	SKIPB (Not allowed at the file level in a spooled file with printer device type *AFPDS.)
EDTWORD	STRPAGGRP
ENDPAGE	TEXT
ENDPAGGRP	TIME
FLTFIXDEC	TIMFMT
FLTPCN	TIMSEP
FONT	TXTRTT
FONTNAME	UNDERLINE
FORCE	ZFOLD
FNTCHRSET	

Performance considerations

- For externally described printer files, the fewer the number of fields in a record, the faster the processing of that record. Also, by putting several lines of text within a record instead of each line as a separate record, system overhead involved with the processing of each record is reduced.
- When coding the DDS for externally described printer files, define the fields in sequential order. The output is not changed if fields are not defined in sequential order, but the extra travel time of the printer head may be noticeable.
- For externally described printer files, specify a specific font or FONT(*CPI) on the CRTPRTF, CHGPRTF, or OVRPRTF command instead of FONT(*DEVD). This helps keep the data stream as small as possible.
- If a spooled file is intended to be printed on an IPDS printer configured AFP(*NO), specify DEVTYPE(*IPDS) on the CRTPRTF, CHGPRTF, or OVRPRTF command to avoid the extra system processing required to transform the data stream from SCS to IPDS.
- If a spooled file is intended to be printed on an IPDS printer configured AFP(*YES), specify DEVTYPE(*AFPDS) on the CRTPRTF, CHGPRTF, or OVRPRTF command.
- When printing on an IPDS printer configured AFP(*YES), large spooled files begin to print sooner if the print while convert (PRTCVT) parameter on the

printer device description is set to *YES. However, some printing may occur before the syntax of the entire spooled file is checked. A data stream error may be found after printing has started. This causes printing to end. If you want all data stream syntax checking to complete before the spooled file starts printing, set the print while convert (PRTCVT) parameter of the printer device description to *NO.

Chapter 3. Spool support

Spooling functions help system users to manage their printing operations more efficiently. This chapter discusses:

- Overview: Spooling
- The elements you need to make spooling work
- Managing spooled files
    - Using the Work with Spooled Files (WRKSPLF) command
    - Restarting and controlling printing
    - Spooled file security
    - Controlling the number of spooled files in your system
    - Redirecting spooled files
    - Copying spooled files
    - Sample commands for additional spooling support

Overview: Why spooling is important

Spooling functions are performed by the system without requiring any special operations by the program that creates the output.

When a program opens a printer file, the operating system, by looking at the printer file SPOOL parameter, determines whether the output is to be spooled.

When a printer file specifying spooling is opened, the spooled file containing the output of the program (data to be printed) is placed on the appropriate output queue in the system. A spooled file can be made available for printing when the printer file is opened, when the printer file is closed, or at the end of the job. This is done by specifying a particular value on the schedule parameter. *IMMED makes the spooled file available to the writer as soon as the program is opened. *FILEEND makes the spooled file available to the writer as soon as the file is closed. *JOBEND makes the spooled file available to the writer as soon as the job is complete.

This process of spooling prevents a potential job limitation imposed by the availability or speed of the printer devices. That is, the system can process application programs that generate printed output much faster than printers can print the output.

By spooling (that is, sending the output to output queues to await printing), the system does not have to wait until the printing for that application program is complete before it can start processing the next application program.

Spooling is especially important in a multiple-user environment where the number of jobs running often exceeds the number of available printer devices. Using spooling, output can be easily redirected from one output queue to another or from one printer to another.

Spooling elements

Following is a list of spooling elements and a diagram showing their working relationship. All of these elements must work together to produce, route, and print spooled files. Following the list is information about each of these elements.

Application program
> A high-level language program that creates a spooled file using a printer file with the SPOOL parameter value set to *YES

Printer file
> A description of the format of the output, and a list of attributes that describe how the system should process the spooled file

Device description
> A description of the printer

Output queue
> An object that contains an ordered list of spooled files to be printed

Printer writer
> A program that takes spooled files from an output queue and sends them to a printer

Remote writer
> A program that takes spooled files from a remote output queue and sends them to a remote system.

Spooled file
> A file containing spooled output records that are to be printed

Figure 1 shows the relationship of these spooling elements.
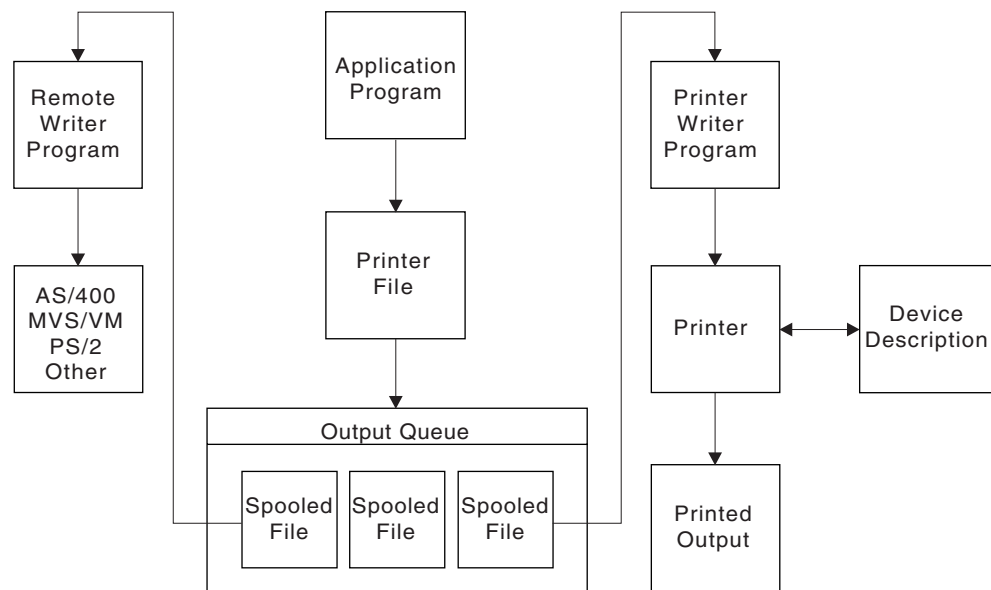
RV3H046-0

Figure 1. Relationship of Spooling Elements

Application program

Application programs are the primary objects that produce spooled files on the iSeries server. All HLL (high-level languages) that can be used on the iSeries server can make use of the spooling support by specifying SPOOL(*YES) in whichever printer file the application uses.

Pressing the Print key runs a program that uses the QSYSPRT printer file to create a spooled file that captures the image that was on the screen when the Print key was pressed.

Printer file

Many attributes of the printer file used by the application program apply to the spooled file and how the spooled file is processed by the system when the application program ends. These attributes can be specified on the following commands:

CHGPRTF
> Change Printer File

CRTPRTF
> Create Printer File

OVRPRTF
> Override with Printer File

After a spooled file is created, its attributes can be displayed using the WRKSPLFA command. Some attributes can be changed using the CHGSPLFA command

Printer device descriptions

Printer descriptions must be created for each printer that is or will be attached to the system. Printer device descriptions are created using the Create Device Description (Printer) (CRTDEVPRT) command or they can also be created automatically by the system if automatic configuration is being used.

Note: Automatic configuration cannot be used for ASCII devices attached to the ASCII workstation controller.

The printer file that an application program uses will require some of the information that is contained in the printer device description. For example: the printer file PRINTER parameter requires the same value as assigned to the printer device description DEVD parameter.

Output queues

Batch and interactive job processing may result in spooled files that are to be sent to a printer. These spooled files contain output records waiting to be printed. There may be many spooled files for a single job.

When a spooled file is created, the file is placed on an output queue. Each output queue contains an ordered list of spooled files. A job can have spooled files on one or more output queues. All spooled files on a particular output queue should have a common set of output attributes, such as printer device, form type, and lines per inch. Using common attributes on an output queue reduces the amount of intervention required and increases the printer throughput.

Output queue security

Output queues are created with a level of security determined by the value of the AUT parameter on the Create Output Queue (CRTOUTQ) command. To work with the spooled files on that output queue, you must have the appropriate authority for that output queue (as specified in the AUT parameter). For example, holding or releasing a spooled file might require one level of authority while reading the contents of that spooled file might require a higher level of authority.

For more information on spooled file and output queue security, see the Security topic in the iSeries Information Center.

Output queue parameters

The following lists the parameters on the Create Output Queue (CRTOUTQ) command and what they specify:

DSPDTA

> Whether users without any special authority, but with *USE authority to the output queue, can display, copy, or send the contents of spooled files other than their own.

JOBSEP

> How many, if any, job separator pages are to be placed between the output of each job and the output of the next job when the output is printed.

OPRCTL

> Whether a user having job control authority can control the output queue (for example, if the user can hold the output queue).

DTAQ The name and library of the data queue associated with an output queue. See "Data queue support on output queues" on page 131 for more information.

MAXPAGES

> The maximum spooled file size, in pages, that is allowed to print between a starting and ending time of day.

LIB The name of the library in which the data queue resides.

SEQ Controls the order in which files are sorted on the output queue. See "Order of spooled files on an output queue" on page 129 for more information.

AUTCHK

> Specifies what type of authority to the output queue is required for a user to control the output queue (for example, to hold the output queue).

AUT Public authority.

RMTSYS

> The name of the remote system to send spooled files to when a remote writer (STRRMTWTR) is started to an output queue.

RMTPRTQ

> The name of the printer queue on the remote system (RMTSYS parameter) that the remote writer sends spooled files to.

AUTOSTRWTR

> The number of remote writers that are automatically started by the system.

MSGQ

> The qualified name of the message queue to which messages are sent when created by a remote writer started to an output queue.

CNNTYPE

The type of connection used to communicate with the remote system.

DESTTYPE

The type of the remote system specified (for example, an iSeries server or a PS/2®).

TRANSFORM

Whether or not to use the host print transform function for SCS, or AFPDS spooled files being sent to a remote system and printed on an ASCII printer.

USRDTATFM

Specifies the name of the data transform program to be used by the driver program.

SEPPAGE

Specifies whether or not to request a separator page when printing on a remote system.

MFRTYPMDL

The manufacturer, type and model for a printer using the host print transform function. This parameter is valid only if TRANSFORM is *YES, or a user data transform program is specified.

WSCST

Name of the workstation customizing object and library. This parameter is valid only if TRANSFORM is *YES, or a user data transform program is specified.

INTNETADR

The name of the Internet address of the remote system.

CLASS

The VM/MVS SYSOUT class for spooled files being sent to a VM/MVS system (*S390 value for the DESTTYPE parameter).

IMGCFG

The name of the image configuration for the output queue. For more information about image configurations. See Chapter 14, "Working with the image print transform function" on page 291.

FCB The name of the forms control buffer for spooled files being sent to a VM/MVS remote system.

DESTOPT

Specifies the destination-dependent options. These options are specific to a particular implementation of an LPD Print Server. When this parameter is not specified, the Send TCP Spooled File (SNDTCPSPLF) command sends only those options that are common to all LPD Print Servers.

When the destination is a Novell server, the text for a banner page can be specified as BANNER='xxxxxxxxxxxx' where xxx can be up to 12 characters. The word Banner must be in upper case, and no blanks should precede or follow the '=' sign.

If you specify *NOWAIT, the remote writer does not wait for the destination system to completely process the spooled file. This is only valid when the Connection Type is *IPX or *SNA.

USRDFNOPT

Specifies one or more user-defined options to be used by user applications or user-specified programs that process spooled files.

USRDFNOBJ

Specifies a user-defined object and type (*DTAARA, *DTAQ, *FILE, *PSFCFG, *USRQ, *USRIDX or *USRSPC) to be used by user applications or user specified programs that process spooled files.

USRDRVPGM

Specifies a user-specified driver program.

SPLFASP

Specifies the auxiliary storage pool (ASP) where the spooled files are to reside.

TEXT   Text description.

Summary of output queue commands

The following commands may be used to create and control output queues. For detailed descriptions of the commands, see the CL Reference topic in the iSeries Information Center.

CHGOUTQ

Change Output Queue: Allows you to change certain attributes of an output queue, such as the sequence of the spooled files on the output queue.

CLROUTQ

Clear Output Queue: Removes all spooled files from an output queue.

CRTOUTQ

Create Output Queue: Allows you to create a new output queue.

DLTOUTQ

Delete Output Queue: Deletes an output queue from the system.

HLDOUTQ

Hold Output Queue: Prevents all spooled files from being processed by the printer writer.

RLSOUTQ

Release Output Queue: Releases a previously held output queue for processing by the printer writer.

WRKOUTQ

Work with Output Queue: Shows the overall status of all output queues, or the detailed status of a specific output queue.

WRKOUTQD

Work with Output Queue Description: Shows descriptive information for an output queue.

Default output queues for printers

When a printer is configured to the system, the system automatically creates the printer's default output queue in library QUSRSYS. The output queue is given a text description of `'Default output queue for printer xxxxxxxxxx'`, where xxxxxxxxxx is the name assigned to the printer during configuration. The printer name is specified in the device description (DEVD) parameter.

The AUT parameter for the output queue is assigned the same value as that specified by the AUT parameter for the printer device description. All other parameters are assigned their default values. Use the Change Command Default (CHGCMDDFT) command to change the default values used when creating output queues with the CRTOUTQ command.

The default output queue for a printer is owned by the user who created the printer device description. In the case of automatic configuration, both the printer and the output queue are owned by the system profile QPGMR.

Default output queues for system printer

The system is shipped with the defaults on commands to use the default output queue for the system printer as the default output queue for all spooled output. The system printer is defined by the QPRTDEV system value.

When a spooled file is created by opening a printer file and the output queue specified for the file cannot be found, the system attempts to place the spooled file on output queue QPRINT in library QGPL. If for any reason the spooled file cannot be placed on output queue QPRINT, an error message is sent and the output is not spooled.

The following output queues are supplied with the system:

QPRINT
> Default printer output queue

QPRINTS
> Printer output queue for special forms

QPRINT2
> Printer output queue for 2-part paper

Creating your own output queues

You can create output queues for each user of the system. For example:

```
CRTOUTQ OUTQ(QGPL/JONES) TEXT('Output queue for Mike Jones')
```

Order of spooled files on an output queue

The order of spooled files on an output queue is mainly determined by the status of the spooled file. A spooled file that is being processed by a writer may have a status of printing (PRT status), writer (WTR status), pending to be printed (PND status), or being sent (SND status). Spooled files with a status of PRT, WTR, PND, or SND are placed at the top of the output queue. A spooled file being processed by the writer may have a held (HLD) status if a user has held the spooled file but the writer is not yet finished processing the file. All other spooled files with a status of RDY are listed on the output queue after the file being processed by a writer, followed by deferred spooled files (DFR status) and spooled files with a status other than RDY or DFR.

Each group of spooled files (RDY and non-RDY files) is further sorted by:

1. The output priority of the spooled file.
2. A date and time field (time stamp).
3. The SCHEDULE parameter value of the spooled file. Files with SCHEDULE(*JOBEND) specified are grouped together and placed after other spooled files of the same job that have SCHEDULE(*IMMED) or SCHEDULE(*FILEEND) specified.
4. The spool number of the spooled file.

For output queues with SEQ(*JOBNBR) specified, the date and time field is the date and time that the job that created the spooled file entered the system. (A sequential job number and time of day value are also assigned to the job when it enters the system.) That is how the spooled files are sorted on the queue.

For first-in-first-out (*FIFO) output queues, the date and time change to the current system date and time when:

- A spooled file is created by opening a device file.
- The output priority of the job that created the spooled file is changed.
- The status of the spooled file changes from non-RDY to RDY.

> **Note:** The date and time do not change when the reason the status changes from RDY to WTR or from WTR to RDY is because the writer was canceled. Also, the date and time don't change when the status changes from RDY to DFR, or from DFR to RDY.

- A spooled file is moved to another output queue that has SEQ(*FIFO) specified.

Because of the automatic sorting of spooled files, different results occur when SEQ(*JOBNBR) is specified for an output queue than when SEQ(*FIFO) is specified. For example, when a spooled file is held and then immediately released on an output queue with SEQ(*JOBNBR) specified, the spooled file will end up where it started; but if the same spooled file were held and then immediately released on an output queue with SEQ(*FIFO) specified, the spooled file would be placed at the end of the spooled files that have the same priority and a status of RDY.

Status of spooled files on an output queue

The status of spooled files, while they are on an output queue, can be determined by running the Work with Output Queue (WRKOUTQ) command and looking at the **STS** column. Following is a list of possible states and a description:

Ready (RDY)
> The spooled file is available to be written.

Open (OPN)
> The spooled file has not been completely processed and is not ready to be selected by a writer.

Closed (CLO)
> The spooled file has been completely processed by a program but SCHEDULE(*JOBEND) was specified and the job that produced the spooled file has not yet finished.

Deferred (DFR)
> The spooled file has been deferred from printing. This status is based on the values assigned to the maximum spooled file size (MAXPAGES) parameter on the CRTOUTQ command. Spooled files will have this status only when a writer is active to the spooled file's output queue.

Held (HLD)
> The spooled file has been held.

Saved (SAV)
> The spooled file has been written and then saved.

Writer (WTR)
> This spooled file is currently being written.

Sending (SND)
> This spooled file is being or has been sent to a remote system. This status only occurs if the remote system (RMTSYS) parameter value on the CRTOUTQ command is something other than *NONE.

Pending (PND)
> This spooled file is pending to be printed.

Printing (PRT)
>   This spooled file has been completely sent to the printer. However, print complete status has not been sent back.

Message waiting (MSGW)
>   This spooled file has a message which needs a reply or an action to be taken.

Controlling printing from an output queue by spooled file size

The maximum spooled file size (MAXPAGES) parameter on the CRTOUTQ command can be used to control printing of spooled files. This is accomplished using three elements of the MAXPAGES parameter:

>   Spooled file size (pages)
>
>   Starting time
>
>   Ending time

Note: The starting and ending times must be specified using the 24 hour method of measuring time. For example, 2:00 PM would be entered as 1400.

Assume you want to restrict spooled files with more than 40 pages from printing between 0800 and 1600 on output queue MYOUTQ. However, between 1200 and 1300 you want to allow spooled files with 10 pages or less to print. Running the following command implements these restrictions:

```
CHGOUTQ OUTQ(MYOUTQ) MAXPAGES((40 0800 1600) (10 1200 1300))
```

Note: You can specify up to 5 different combinations of pages and start/stop times on the MAXPAGES parameter.

Data queue support on output queues

Support is available to optionally associate a data queue with an output queue using the Create Output Queue (CRTOUTQ) or Change Output Queue (CHGOUTQ) command. Entries are logged in the data queue when spooled files are in ready (RDY) status on the output queue. A user program can determine when a spooled file is available on an output queue using the Receive Data Queue API (QRCVDTAQ) to receive information from a data queue. See the API Programming book for more information about data queues and the QRCVDTAQ API.

Each time a spooled file on the output queue reaches RDY status an entry is sent to the data queue. A spooled file can have several changes in status (for example, ready (RDY) to held (HLD) to release (RLS) to ready (RDY) again) before it is taken off the output queue. These status changes result in entries in the data queue for a spooled file each time the spooled file goes to RDY status.

A spooled file can reach RDY status:
* When initially spooled on the output queue.
* When the spooled file is opened and the schedule parameter value is *IMMED.
* When a job completes and the spooled file schedule parameter value is *JOBEND.
* When the spooled file is released.
* When a spooled file is moved to this output queue from another output queue.
* When a writer is ended immediately while printing a spooled file (the spooled file status is reset from WTR to RDY).

Environment variable QIBM_NOTIFY_CRTSPLF data queue support

Using the ADDENVVAR or CHGENVVAR command, you can associate a data queue with a job or the system. As spooled files are created, the entries are logged in the data queue. Using the Receive Data Queue API (QRCVDTAQ) to receive information from the data queue, a user program can determine when a spooled file has been created by the job or by the system.

Specifying data queues through the environment variable QIBM_NOTIFY_CRTSPLF: Using the CL command ADDENVVAR, and specifying a fully-qualified data queue name for the environment variable QIBM_NOTIFY_CRTSPLF, you can associate a data queue with a job or the system.

The command use would be:

```
ADDENVVAR ENVVAR(QIBM_NOTIFY_CRTSPLF)
          VALUE('*DTAQ <library name>/<data queue name>')
          LEVEL(*JOB | *sys)
```

Once a data queue is associated with a job or the system, any spooled file created by the job or system will automatically have an entry placed in the data queue. For this action to occur, the user or user profile QSPL must have authorization to the data queue. For more information about data queues, see"Creating data queues".

Note: An environment variable that is specified at the job level takes precedence over the same environment variable specified at the system level.

Creating data queues

The Create Data Queue (CRTDTAQ) command is used to create the data queue. The maximum message length (MAXLEN) parameter value should be specified as at least 128 for a spool data queue entry record type of 01. The sequence (SEQ) parameter value should be *FIFO or *LIFO.

The data queue for the spool data queue entry record type of 02, must be created with a record length of at least 144 bytes. It must also have a public authority of *USE, or grant QSPL user profile *USE private authorities to the data queue. You must ensure that the containing library has a public authority of *EXECUTE, or grant QSPL user profile *EXECUTE private authorities to the library. The format of the CRTDTAQ command is:

```
CRTDTAQ DTAQ (<library name>/<data queue name>) MAXLEN(144) AUT(*USE)
```

Specifying data queues on the CRTOUTQ and CHGOUTQ commands

The CRTOUTQ and CHGOUTQ commands have a data queue (DTAQ) parameter, which is used to specify the data queue name. An error occurs when using these commands if the specified data queue does not exist or if the user creating or changing the output queue does not have use authority to the data queue.

After a data queue is associated with an output queue, any spooled file that is placed on the output queue in ready status causes an entry to be placed on the data queue. The data queue entry is added regardless of the authority the user generating the spooled file has to the data queue.

Working with data queues when errors occur

If the iSeries server tries to add entries to a data queue that does not exist or has an invalid length, the system continues with its processing but sends an informational message to the QSYSOPR message queue. This message indicates

that there is a problem with the data queue and specifies the data queue name. This message is sent the first time a specific problem occurs with the data queue of an output queue. The message is sent once every 24 hours.

For example, if message X is received at 10:00 AM, it is logged in the QSYSOPR message queue. If message X is received again at 10:30 AM, 11:00 AM, 1:00 PM, or 1:30 PM, it will not be logged. As you can see, the message will not be logged until after 10:00 AM the next day, even if it continues to be received all day.

If after message X is logged at 10:00 AM, message Y is received at 2:00 PM, message Y is logged. If message X is received again at 2:30 PM, message X will be logged again even though it was logged earlier in the day.

The intent is not to log the same recurring message all day, but to inform the user of each change of error messages associated with the data queue of a particular output queue.

Managing data queue entries

Changing the data queue of an output queue is allowed regardless of whether there are spooled files on the output queue. For data queue entries of record type 01, only spooled files that reach RDY status after the change will have entries on the data queue. Spooled files already having a status of ready on the output queue will not have entries on the new data queue.

It is the user's responsibility to manage the data queues. These responsibilities include creating, clearing, and deleting data queues.

When clearing all output queues during IPL, any associated data queues are not cleared. If a damaged system output queue is found, it is re-created without any associated data queue name. Damaged data queues are not re-created.

Record type 01 data queue entry format

Following is the format of a 01 data queue entry when a spooled file changes to ready status on an output queue.

Table 6. Format of data queue entry for a spooled file

Decimal Offset	Hex Offset	Type	Description
0	0	CHAR(10)	Function Identifies the function that created the data queue entry. The value for a spooled file is *SPOOL.
10	A	CHAR(2)	Record type Identifies the record type within the function. Valid values are: **01** A spooled file that is in READY status has been placed on the output queue.

Table 6. Format of data queue entry for a spooled file (continued)

Decimal Offset	Hex Offset	Type	Description

12	C	CHAR(26)	Qualified job name
			
			Identifies the qualified job name of the job that created the spooled file placed on the output queue.
			
			CHAR(10)
			        Job name
			
			CHAR(10)
			            User name
			
			CHAR(6)
			        Job number

38	26	CHAR(10)	Spooled file name Identifies the name of the spooled file placed on the output queue.
48	30	BINARY(4)	Spooled file number Identifies the unique number of the spooled file placed on the output queue.

52	34	CHAR(20)	Qualified output queue name
			
			Identifies the qualified name of the output queue on which the spooled file was placed.
			
			CHAR(10)
			            Output queue name
			
			CHAR(10)
			            Library of the output queue

72	48	CHAR(8)	Job system name. Identifies the name of the system on which the spooled file was generated.
80	50	CHAR(7)	Spooled file create date. Identifies the date on which the spooled file was created in CYYMMDD format.
87	57	CHAR(1)	Reserved
88	58	CHAR(6)	Spooled file create time. Identifies the time that the spooled file was created in HHMMSS format.
94	5E	CHAR(34)	Reserved

Record type 02 data queue entry format

Following is the format of a DTAQ entry for creating a spooled file.

Table 7. Format of a data queue entry for creating a spooled file

Decimal Offset	Hex Offset	Type	Description
0	0	CHAR(10)	Function Identifies the function that created the data queue entry. The value for a spooled file is *SPOOL.

Decimal Offset	Hex Offset	Type	Description
10	A	CHAR(2)	Record type Identifies the record type within the function. Valid values are: **02** A spooled file has been created and placed on the output queue.

12	C	CHAR(26)	Qualified job name
			
			Identifies the qualified job name of the job that created the spooled file placed on the output queue.
			
			CHAR(10)
			    Job name
			
			CHAR(10)
			    User name
			
			CHAR(6)
			    Job number

38	26	CHAR(10)	Spooled file name Identifies the name of the spooled file placed on the output queue.
48	30	BINARY(4)	Spooled file number Identifies the unique number of the spooled file placed on the output queue.

52	34	CHAR(20)	Qualified output queue name
			
			Identifies the qualified name of the output queue on which the spooled file was placed.
			
			CHAR(10)
			    Output queue name
			
			CHAR(10)
			    Library of the output queue

72	48	CHAR(26)	Creating qualified job name
			
			Identifies the qualified job name of the job that created the spooled file.
			
			CHAR(10)
			    Job name
			
			CHAR(10)
			    User name
			
			CHAR(6)
			    Job number

98	62	CHAR(10)	User data Identifies the user specified data for the spooled file that was created.

Table 7. Format of a data queue entry for creating a spooled file (continued)

Decimal Offset	Hex Offset	Type	Description
108	6(C)	BINARY(4)	Thread ID Identifies the thread of the job that created the spooled file.
112	70	CHAR(10)	System name Identifies the name of the system on which the spooled file was generated.
122	7A	CHAR(7)	Creation date Identifies the date on which the spooled file was created in CYYMMDD format..
129	81	CHAR(6)	Creation time Identifies the time that the spooled file was created in HHMMSS format.
135	87	CHAR(9)	Reserved

Using multiple output queues

You may want to create multiple output queues for:

- Special forms printing
- Output to be printed after normal working hours
- Output that is not printed

  An output queue can be created to handle spooled files that need only to be displayed or copied to a database file. Care should be taken to remove unneeded spooled files.

- Special uses

  For example, each programmer could be given a separate output queue.

- Output of special IBM files

  You may want to consider separate queues for the following IBM-supplied files:

  – QPJOBLOG: You may want all job logs sent to a separate queue.
  – QPPGMDMP: You may want all program dumps sent to a separate queue so you can review and print them if needed or clear them daily.
  – QPSRVDMP: You may want all service dumps sent to a separate queue so the service representative can review them if needed.

Controlling multiple output queues

Controlling multiple output queues requires both finding where your output is and determining how to print it if a writer is not started to the queue.

The Work with Spooled Files (WRKSPLF) command can be used to display all the spooled files that you have created. This is an easy way to find your output if you do not know the name of the output queue where it has been placed. The spooled files are listed in the same order as they would be on a SEQ(*FIFO) output queue. (See "Order of spooled files on an output queue" on page 129 for more information.)

If you know the name of the output queue that contains the spooled file, the Work with Output Queue (WRKOUTQ) command can be used to display the queue to determine the position of the spooled files on that output queue.

The WRKOUTQ command also provides the option to display all the output queues that exist on your system.

If a writer is not started to the output queue that contains the spooled file you wish to print, you have several options in order to print the file:
- Use the Work with All Spooled Files (WRKSPLF) command and type 9 (Work with printing status) next to your spooled file to find out Why it is not printing.
- Use the Change Spooled File Attributes (CHGSPLFA) command to move the spooled file to an output queue that has a writer started to it.
- Select a printer that is not being used and use the Start Printer Writer (STRPRTWTR) command to start a printer that will print the spooled files from your output queue.
- Select a printer that is started to a different output queue and use the Change Writer (CHGWTR) command to change the printer to print the spooled files from your output queue.

If a writer is started to the output queue that contains the spooled file you want to print, but the status of the spooled file is deferred (DFR status), you can do the following:
- Use the Change Spooled File Attributes (CHGSPLFA) command to move the spooled file to an output queue that has a value of *NONE specified for the maximum spooled file size (MAXPAGES) parameter.
- Use the Change Writer (CHGWTR) command to change the printer to print the spooled files from the output queue that has the MAXPAGES parameter value specified as *NONE.

The Work with Writers (WRKWTR) command can be used to find a list of printers attached to your system and to determine if the printer is active (started).

Output queue recovery
If a job that has produced spooled files is running when the job or system stops abnormally, the files remain on the output queue. Some number of records written by active programs may still be in main storage when the job ends and will be lost. You should check these spooled files to ensure that they are complete before you decide to continue using the files.

You can use the SPLFILE parameter on the End Job (ENDJOB) command to specify if all spooled files (except QPJOBLOG) created by the job are to be kept for normal processing by the printer writer, or if these files are to be deleted.

If an abnormal end occurs, the spooled file QPJOBLOG will be written at the next IPL of the system.

If a printer writer fails while a spooled file is being printed, the spooled file remains on the output queue intact.

Recovery of user-created output queues: If an output queue becomes damaged in such a way that it cannot be used, you are notified by a message sent to the system operator message queue. The message comes from a system function when a printer writer or a job tries to put or take spooled files from the damaged queue.

A damaged output queue can be deleted using the Delete Output Queue (DLTOUTQ) command, or it will be deleted by the system during the next IPL.

After a damaged output queue is deleted, all spooled files on the damaged output queue are moved to output queue QSPRCLOUTQ in library QRCL. The move is performed by the QSPLMAINT system job, which issues a completion message to the QSYSOPR message queue when all spooled files have been moved.

After the damaged output queue is deleted, it can be created again by entering the Create Output Queue (CRTOUTQ) command. Then, spooled files on output queue QSPRCLOUTQ can be moved to the newly created output queue using the Change Spooled File Attributes (CHGSPLFA) command.

Recovery of system-created output queues: If the output queue that was damaged was the default output queue associated with a printer, the system automatically re-creates the output queue when it is deleted.

This system-created output queue has the same public authority as specified for the device and default values for the other parameters. After the system re-creates the output queue, you should verify its attributes are correct or change them, if necessary, using the Change Output Queue (CHGOUTQ) command.

When a damaged output queue associated with a printer is deleted and created again, all spooled files on the damaged queue are moved to the re-created output queue. This is done by the QSPLMAINT system job, which issues a completion message to the QSYSOPR message queue when all spooled files have been moved.

Spooled file cleanup after an abnormal IPL
System spool cleanup starts immediately following an abnormal IPL. Spool cleanup is done under the system job QSPLMAINT.

Spooled files on destroyed user-created output queues are moved to output queue QSPRCLOUTQ in library QRCL. Spooled files on destroyed system-created output queues are moved to the re-created output queues.

Cleanup also deletes spooled files that have data in a damaged database file in library QSPL.

Printer writer program

A printer writer is an OS/400 program that takes spooled files from an output queue and sends them to a printer. The spooled files on a particular output queue remain stored in the system until the printer writer program assigns a printer to the output queue.

More than one printer writer can be started to the same output queue (10 is the limit). However, each writer name must be unique and of the same type (printer, remote, or diskette).

The printer writer program takes spooled files, one at a time, from the output queue, based on their priority. The printer writer program prints a spooled file only if its entry on the output queue indicates that it has a ready (RDY) status. You can display the status of a particular spooled file using the Work with Output Queue (WRKOUTQ) command.

If the spooled output file has a ready status, the printer writer program takes the entry from the output queue and prints the specified job and/or file separators,

followed by the output data in the spooled file. If the spooled file does not have a ready status, the printer writer program leaves the entry on the output queue and goes on to the next entry. In most cases the printer writer program continues to print spooled files (preceded by job and file separators) until all spooled files with a ready status have been taken from the output queue.

Notes:

1. The printer writer program uses the printer file QPSPLPRT. This printer file is shipped with the system. It is set up for the printer writer program and should not be changed or used for other applications.

2. If you run the CHGPRTF command to make all the IBM-supplied printer files DBCS capable (CHGPRTF FILE(*all/*all) IGCDTA(*YES)), you must change the IGCDTA parameter value for QPSPLPRT printer file back to *NO.

The AUTOEND parameter on the Start Printer Writer command determines whether the printer writer program continues to wait for new spooled files to become available to be printed, ends after printing one file, or ends after all spooled files with ready status have been taken from the output queue.

The SEPDRAWER parameter on the Start Printer Writer (STRPRTWTR) and Change Writer (CHGWTR) commands allows the job separators and file separators to be printed on paper selected from a different drawer. Typically this is used to print the separators on colored paper.

The INIT parameter on the STRPRTWTR command allows you to specify when to initialize (send printer open time commands) to the printer..

The FORMTYPE parameter on the STRPRTWTR command allows several values for the message option:

*INQMSG
> This is the default value.
>
> An inquiry message is issued if the spooled file has a form type that is different from what is in the printer.

*INFOMSG
> An informational message is issued when no spooled files of the specified form type remain on the output queue.

*MSG Both the inquiry and informational messages are issued.

*NOMSG
> No messages are issued.

These values are also supported on the Change Writer (CHGWTR) command.

Remote writer program

A remote writer is an OS/400 program that takes spooled output files from a remote output queue and sends them to the specified remote system. The remote writer, which is a system job, sends the spooled output files using SNADS or TCP/IP. This function is known as remote system printing on the iSeries server. The Start Remote Writer (STRRMTWTR) command is used to initiate remote system printing.

After the spooled output file is successfully sent to a remote system, it is deleted or saved as determined by the SAVE spooled file attribute value.

More than one remote writer can be started to the same remote output queue (10 is the limit). The actual number is specified in the remote output queue description. However, each writer name must be unique and of the same type (printer, remote, or diskette). See Chapter 4, "Remote System Printing" on page 159 for more information on remote writers.

Using multiple printer writer support

Multiple printer writers can be started to one output queue. The limit is 10. This support allows many printers (up to 10) to begin printing spooled output files from the same output queue.

The multiple printer writer function supports work load balancing between printers. It also provides backup for printed jobs running unattended. For example, if one printer jams or runs out of paper the others continue to print spooled output files from the associated output queue.

Locating your printed output

The user profile parameter, user options, has a value that allows notification of when spooled files are sent, held, or printed. This value is *PRTMSG. The notification is returned in various messages. The message number is different depending on which writer command (Start Remote Writer (STRRMTWTR) or Start Printer Writer (STRPRTWTR) command) was used.

When working with the Start Printer Writer (STRPRTWTR) command, message CPI34B8 indicates that the spooled file printed successfully and on which printer it printed. Message CPD34B9 indicates that the spooled file is being held. CPD34B9 also indicates which printer it would have printed on.

When working with the Start Remote Writer (STRRMTWTR) command, message CPI34B7 indicates that the spooled file was sent successfully. Message CPD34B7 indicates the spooled file was not successfully sent.

Summary of writer commands for printing

The following commands may be used to work with the printer writer program. For detailed descriptions of the commands, see the CL Reference topic in the iSeries Information Center.

STRPRTWTR

> Start Printer Writer: Starts the printer writer program to assign an active printer to an output queue. Spooled files on that output queue are printed on the assigned printer.

> This command also allows you to specify the message queue for printer writer messages, which form types should be printed, number of file separator pages, and the drawer for separator pages.

STRRMTWTR

> Start Remote Writer: Starts a remote writer program to send spooled output files from a remote output queue to a remote system.

> This command also allows you to specify the message queue for remote writer messages, which form types should be printed, and whether the writer ends automatically.

CHGWTR

> Change Writer: Allows you to change some printer writer program attributes, such as form type, drawer for separator pages, number of file separator pages, or output queue.

HLDWTR

Hold Writer: Temporarily stops a printer writer program at the end of a record, at the end of a file, or at the end of a page.

RLSWTR

Release Writer: The printer writer program continues from the point at which it was previously held.

ENDWTR

End Writer: Ends a printer writer program and makes the associated printer device available to the system.

Using a printer for both spooled files and direct print jobs

The allow direct print function provides the capability of sharing a printer between the printer writer and direct print jobs.

A printer writer sends spooled files to the printer from the output queue assigned to that printer. When the SPOOL parameter value on the printer file is *YES, output data is written to a spooled file and placed on an output queue.

When the SPOOL parameter value on the printer file is *NO, output data is written directly to the printer. This is a direct print job.

Enabling the allow direct print function

To allow direct print jobs as well as spooled files for a printer, specify *YES as the value on the allow direct print (ALWDRTPRT) parameter on the Start Printer Writer (STRPRTWTR) command.

When *YES is specified, the printer writer releases the printer for direct print jobs when either of the following conditions is true:

- The printer writer is held (STATUS(*HLD)).
- The output queue associated with the printer contains no spooled files waiting to print.

The allow direct print function uses the maximum file-wait time specified in the WAITFILE parameter in the printer file.

The default value for the WAITFILE parameter is *IMMED. However, the allow direct print function requires some time to obtain access to the printer. Therefore, it is recommended that 2 or 3 minutes be allowed for access to the printer.

Note: The WAITFILE parameter value must be calculated in seconds. For example, a parameter value of 120 would allow a wait time of 2 minutes.

When choosing the value for the maximum file-wait time (WAITFILE) parameter in the printer file, consider how busy is the printer that you want to use for your direct print jobs?

If the value for the WAITFILE parameter is too small, a message is returned indicating the specified printer could not be obtained. If you still need the job printed, the job must be run again.

If the value for the WAITFILE parameter is too large, your sign-on session is utilized waiting for the job to obtain the printer. You cannot use System Request to

end the direct print job. The job must be ended using the End Job (ENDJOB) command. Using this command ends your sign-on session.

Modifying the start printer writer (STRPRTWTR) command

The STRPRTWTR command can be modified to allow all printers being started to accept direct print jobs. Use the Change Command Default (CHGCMDDFT) command and set the value for the allow direct print (ALWDRTPRT) parameter on the STRPRTWTR command to *YES. Thereafter, when the STRPRTWTR command is used, the allow direct print function is enabled.

Restrictions when using the allow direct print function

The Change Writer (CHGWTR) command cannot be used to enable the allow direct print function.

The allow direct print function is not supported on printers configured for advanced function printing. This means the advanced function printing (AFP) parameter on the printer device description must be *NO.

Managing spooled files

When jobs that produce spooled files are running, they use specified printer files to determine where the spooled files will go and what they will look like.

The spool support on the iSeries server allows you to change the destination of spooled files and to alter the appearance of a spooled file before actual printing occurs. The Work with Spooled Files (WRKSPLF) command is the primary command used for managing your spooled files.

Using the work with spooled files (WRKSPLF) command

By typing WRKSPLF on a command line you can display a list of your spooled files similar to the example below.

The Work with All Spooled Files display shows you, in the *Device or Queue* column, the name of the queue that the spooled file is currently in.

At the top of the display is a list of actions that can be performed on any spooled file.

All of these actions (with the exception of Messages and Work with printing status) can be performed using CL commands. The Work with All Spooled Files display provides a convenient way for you to run these commands. When you use this display, you do not have to remember the exact CL command name and the spooled file attributes such as: name, number, and position in a particular output queue.

If you are using the WRKSPLF command, you can manage your spooled files by selecting one of the available options from the following list. This list provides the option, the associated CL commands, and an explanation of each option.

- Send — Send Network Spooled File (SNDNETSPLF) command

  This option allows you to send the spooled file to another user on your system or to a user in your network. Go to "Using the send option" on page 143 for more information on using this option.

- Change — Change Spooled File Attributes (CHGSPLFA) command

  This option allows you to change attributes of a spooled file.

- Hold — Hold Spooled File (HLDSPLF) command

This option allows you to stop the processing of a spooled file by a printer writer.

- Delete — Delete Spooled File (DLTSPLF) command

    This option allows you to remove the spooled file from the system.

- Display — Display Spooled File (DSPSPLF) command

    This option allows you to display the contents of a spooled file.

- Release — Release Spooled File (RLSSPLF) command

    This option allows the printer writer to process a previously held or saved spooled file.

- Messages — No associated CL command

    This option displays any messages associated with the spooled file.

- Attributes — Work with Spooled File Attributes (WRKSPLFA) command

    This option allows you to change attributes that control how a spooled file will look when it is printed. Go to "Using the attributes option" on page 145 for examples of changing these attributes.

- Work with printing status — Work with Printing Status (WRKPRTSTS) command

    This option allows you to view the conditions that are preventing the spooled file from printing. For examples on how to use this option, go to "Using the work with printing status option" on page 147.

Using the send option
Selecting option 1 (Send) from the Work with All Spooled Files display allows you to send a spooled file to another user on the system or to a user in a communications network that your iSeries server is linked to.

Notes:

1. The users to whom you want to send the spooled file must be in the system directory. Also, if the user to whom you want to send the spooled file is a user on another system, the name of that system must be in the system directory. Use the Display Directory (DSPDIR) command to view the list of users and systems in your communications network.

2. In most cases you will want to specify *ALLDATA as the value for the *Data format* prompt. This will ensure that all the attributes of the spooled file are sent with the file.

By choosing option 1 from the WRKSPLF display, the only information you have to type to send the spooled file is the user ID and address.

Note: If your system is in a TCP/IP network, you can send and print spooled files using the Send TCP/IP Spooled File (SNDTCPSPLF) command. For more information, see "Sending and printing files with TCP/IP" on page 324.

Using the messages option
Selecting option 7 (Messages) from the Work with All Spooled Files display allows you to view any system messages concerning a particular spooled file.

If a spooled file does not start to print or finish printing, the reason could be that a message did not receive a response. Often, it is a check to make sure that the forms have the correct alignment or position in the printer. Answering these messages allows the job to start or continue printing.

Using the change option

Selecting option 2 (Change) from the Work with All Spooled Files display allows you to change certain attributes of a spooled file. Below is a list that indicates which of the spooled file attributes you can change using option 2.

Notes:

1. To find out the possible values you could specify for any of these attributes, move the cursor under any of the attributes and press the Help key.

2. If the spooled file is already printing (status is WTR), only a subset of this list of attributes can be changed.

The attributes included in the subset vary depending on the device type (*IPDS, *SCS, *AFPDSLINE, *LINE, and *AFPDS).

```
Printer
Print sequence
Form type
Copies
Restart printing
Output queue
 Library
File separators
Page range to print:
 Starting page
 Ending page
File becomes available
Save file
Output priority
User data
Align page
Print quality
Form feed
Source drawer
Print fidelity
Print on both sides
Form definition
 Library
AFP characters
Pages per side
Page definition
 Library
Front side overlay:
 Overlay
 Library
  Offset down
  Offset across
Back side overlay:
 Overlay
 Library
  Offset down
  Offset across
Constant back
User defined object:
 Object
 Library
  Object type
IPDS pass-through
Font resolution
```

Example: To change the specified printer: Use option 2 from the Work with All Spooled Files display by typing 2 next to the spooled file you want to work with.

The Change Spooled File Attributes display appears.

Type the name of the printer you want to specify for this spooled file.

Press the Enter key and you are returned to the Work with All Spooled Files display.

You have now changed the specified printer for that spooled file.

Restrictions to changing spooled file attributes: Some attributes of a spooled file cannot be changed based on the device type (DEVTYPE) of the spooled file.

The pages per side (MULTIUP), front overlay (FRONTOVL), and back overlay (BACKOVL) attributes can only be changed if the DEVTYPE is *SCS or *IPDS.

The above attributes plus form feed (FORMFEED), print quality (PRTQLTY), and print on both sides (DUPLEX) cannot be changed on spooled files created on the iSeries server with printer device type *AFPDS.

Using the attributes option
Selecting option 8 (Attributes) from the Work with All Spooled Files display allows you to view a list of all the attributes of the specified spooled file.

You can also use the Work with Spooled File Attributes (WRKSPLFA) command to change the attributes of a spooled file.

Initially, the attributes assigned to a spooled file are obtained from the printer file that the application program used.

You cannot change all of the attributes that are shown. To view the changeable attributes, press F13.

You can also change the attributes in the Using the change option section by pressing F13.

Note: The attributes included in the subset vary depending on the device type (*IPDS, *SCS, *USERASCII, *AFPDSLINE, *LINE, and *AFPDS).

Device requirements
When you work with spooled file attributes, the device requirements shown in the following table are displayed. The table shows the device requirements supported on each printer.

Table 8. Supported Device Requirements

Device Requirement	Printer											
	3287 4245 5256 5262 6262	5252	4214	5224 5225	4234-2	5219	3812 3816 SCS	5553[1]	3112 3116 3812 3816 3916 3930 4028 4312 4317 4324 Info Print 20 Info Print 32 IPDS	4230 4224 4234-8 4234-12 4247 6400 6408 6412	5583	3130 3160 3820[1] 3825[1] 3827[1] 3829[1] 3831[1] 3835[1] 3900[1] 3935[1] Info Print 3000 Info Print 4000
Final Form Text			X			X	X	X	X	X		X
Variable Font						X	X	X	X	X		X
Variable LPI			X		X	X	X	X	X	X	X	X
Variable Drawer			X			X	X	X	X	X	X	X
Super/Subscript						X	X	X	X	X		X
Variable Character ID			X	X	X	X	X	X	X	X		X
Highlight							X	X	X	X		X
Extended 3812 Fonts							X	X	X	X		X
Graphics 522X				X	X				X			X
Graphics 4214			X						X			X
Graphics 4234					X				X			X
Graphics									X	X		X
Barcodes									X	X		X
Variable Page Rotation							X	X	X			X
PC Printer Emulation	X	X	X	X	X	X	X	X			X	
Defined Characters			X	X	X				X			X	X
Variable CPI		X	X	X	X	X			X			X	X

Table 8. Supported Device Requirements (continued)

	Printer											
Device Require-ment	3287 4245 5256 5262 6262	5252	4214	5224 5225	4234-2	5219	3812 3816 SCS	5553[1]	3112 3116 3812 3816 3916 3930 4028 4312 4317 4324 Info Print 20 Info Print 32 IPDS	4230 4224 4234-8 4234-12 4247 6400 6408 6412	5583	3130 3160 3820[1] 3825[1] 3827[1] 3829[1] 3831[1] 3835[1] 3900[1] 3935[1] Info Print 3000 Info Print 4000
Transparency			X	X	X			X			X	X
IPDS Transparent Data								X	X	X		X
Field Outlining								X			X	X
AFP Resources									X	X		X
Rotated DBCS Characters								X			X	X
Double-wide Characters								X			X	X

[1] System does not check device attributes for these printers, so all attributes are permitted. This does not guarantee the spooled file will print correctly.

Using the work with printing status option

Selecting option 9 (Work with printing status) from the Work with All Spooled Files display helps you find the reason Why a spooled file is not printing.

Type a 9 (Work with printing status) next to the spooled file that is not printing.

The conditions that may be preventing the spooled file from printing are displayed.

Type a 2 (Change status) next to the condition you want to eliminate. With the information shown in the Change Status window, you can take corrective action to make the spooled file print. All of the conditions listed must be eliminated before the spooled file will print.

If you need more information to eliminate the conditions, select option 5 (Display detailed description) for an explanation of each of the conditions preventing the spooled file from printing.

Restarting and controlling printing

In some cases, you may want to restart printing or control a spooled file while it is being printed. For example:

- The system ended while a spooled file was being printed, or you want to print only selected portions of a large spooled file.

  You can use the Work with Spooled Files (WRKSPLF) command, the Work with Output Queue (WRKOUTQ) command, or the Work with Job (WRKJOB) command to see a list of spooled files. Type 2 (Change) next to the spooled output file you want to work with and press the Enter key. The Change Spooled File Attributes display appears.

  Locate the *Restart printing* parameter and type *STRPAGE. Locate the *Page range to print* parameter. This parameter has two parts: *Starting page* and *Ending page*. Type the page number that you want the spooled file to start or resume printing on and type the page number that you want that spooled file to stop printing on.

- A spooled file needs to be printed immediately, but another file is currently printing.

  For example, assume spooled file A is currently printing. While A is printing, you receive a request to have spooled file B printed immediately.

  Use the Work with Spooled Files (WRKSPLF) command or Work with Output Queue (WRKOUTQ) command to locate spooled files A and B.

  Next to B, type 2 (Change) and press the Enter key. The Change Spooled File Attributes display appears. Locate the *Print sequence* parameter. Change the value of the *Print sequence* parameter to *NEXT and press the Enter key. This moves B to the top of the output queue.

  Next to spooled file A (currently printing and having a status of WTR), type 3 (Hold), move the cursor to the command line, and type OPTION(*PAGEEND). Press the Enter key and A stops printing at the end of the current page.

  This allows spooled file B to start printing.

  When you are ready to finish printing spooled file A, type 2 (Change) next to A and press the Enter key.

  The Change Spooled File Attributes display appears. Locate the *Print sequence* parameter. Change the value of the *Print sequence* parameter to *NEXT. Locate the *Restart printing* parameter and change the value to *NEXT. File A is released and will be the next spooled file to print, and the system will resume printing with the page following the last page printed.

- You want to be notified when a spooled file completes printing or is held by the printer writer program.

  You can specify *PRTMSG for the user option in your user profile to tell the system to notify you when your spooled file is being held by the printer writer or when printing is complete. Use the Create User Profile (CRTUSRPRF) command, the Change User Profile (CHGUSRPRF) command, or the Change Profile (CHGPRF) command, and look for the *User option* prompt (USROPT parameter). See the Security topic in the iSeries Information Center for more information about these commands and the *PRTMSG value of the USROPT parameter.

Spooled file security

Spooled security is primarily controlled through the output queue that contains the spooled files. In general, there are four ways that a user can become authorized to control a spooled file (for example, hold or release the spooled file):

- User is assigned spool control authority (SPCAUT(*SPLCTL)) in the user profile. This authority gives a user control of all spooled files in the output queues of all libraries to which the the user has *EXECUTE authority. This authority should only be granted to appropriate users.
- User is assigned job control authority (SPCAUT(*JOBCTL)) in the user profile, the output queue is operator-controlled (OPRCTL(*YES)), and the user has *EXECUTE authority to the library that the output queue is in.
- User has the required object authority for the output queue. The required object authority is specified by the AUTCHK parameter on the CRTOUTQ command. A value of *OWNER indicates that only the owner of the output queue is authorized to control all the spooled files on the output queue. A value of *DTAAUT indicates that users with *CHANGE authority to the output queue are authorized to control all the spooled files on the output queue.

  Note: The specific authorities required for *DTAAUT are *READ, *ADD, and *DLT data authorities.
- A user is always allowed to control the spooled files created by that user.

For the Copy Spooled File (CPYSPLF), Display Spooled File (DSPSPLF), and Send Network Spooled File (SNDNETSPLF) commands, in addition to the four ways already listed, there is an additional way a user can be authorized.

If DSPDTA(*YES) was specified when the output queue was created, any user with *USE authority to the output queue is allowed to copy, display, send, or move spooled files. The specific authority required is *READ data authority.

If the user is authorized to control the file by one of the four ways already listed above, using DSPDTA(*NO) when creating the output queue will not restrict the user from displaying, copying, or sending the file. DSPDTA authority is only checked if the user is not otherwise authorized to the file.

DSPDTA(*OWNER) is more restrictive than DSPDTA(*NO). If the output queue is created with DSPDTA(*OWNER), only the owner of the spooled file (the person who created it) or a user with SPCAUT(*SPLCTL) may display, copy, or send a file on that queue. Even users with SPCAUT(*JOBCTL) on an operator-controlled (OPRCTL(*YES)) output queue cannot display, copy, move, or send spooled files they do not own.

See the See the Security topic in the iSeries Information Center for details about the authority requirements for individual commands.

To place a spooled file on an output queue, one of the following authorities is required:
- Spool control authority (SPCAUT(*SPLCTL)) in the user profile. The user must also have the *EXECUTE authority to the library that the output queue is in.

  This authority gives a user control of all spooled files on the system and should only be granted to appropriate users. If you have spool control authority you can delete, move, hold, and release any spooled files on the system. You can also change the attributes of any spooled file.
- Job control authority (SPCAUT(*JOBCTL)) in the user profile and the output queue is operator-controlled (OPRCTL(*YES)). The user must also have the *EXECUTE authority to the library that the output queue is in.
- *READ authority to the output queue. This authority can be given to the public by specifying AUT(*USE) on the CRTOUTQ command.

Controlling the number of spooled files in your system

The number of spooled files in your system should be limited. When a job is completed, spooled files and internal job control information are kept until the spooled files are printed or canceled. The number of jobs on the system and the number of spooled files known to the system increase the amount of time needed to perform initial program load (IPL) and internal searches, and increase the amount of temporary storage required.

The number of jobs known to the system can be displayed using the Work with System Status (WRKSYSSTS) command.

You can use the Work with Spooled Files (WRKSPLF) command to identify spooled files that are no longer needed. By periodically entering the command:

```
WRKSPLF SELECT(*ALL)
```

you can determine which spooled files are older than 6 or 7 days, and then delete the spooled files or contact the users who created them.

For detailed information on minimizing the number of job logs (for example, by using LOG(4 0 *NOLIST)), see the Control Language topic in the iSeries Information Center. For information regarding the use of system values to control the amount of storage associated with jobs and spooled files, see the Work Management topic in the iSeries Information Center.

Reclaiming empty spooled file members

When a spooled file has been deleted, all of the spooled file data has been erased. However, an empty spooled file member still resides in auxiliary storage waiting to be reused.

Having some empty spooled file members available for creating new spooled files increases system run-time performance. However, a large number of empty spooled files can use large amounts of storage and decrease system abnormal IPL performance. For example, each spooled file member may take 24k of storage.

You can determine the balance you want to maintain between spool performance and auxiliary storage use by setting the system value Reclaim Spool Storage (QRCLSPLSTG) to a certain number of days. The default value for QRCLSPLSTG is 8 days.

As an alternative, you can run the Reclaim Spool Storage (RCLSPLSTG) command with the DAYS parameter set to *NONE to immediately reclaim all empty spooled file members.

Notes:
1. System performance is degraded if QRCLSPLSTG is run and the DAYS parameter value is 0.
2. The database member is immediately deleted after the deletion of a spooled file. That means that there is no pool of unused members that can be used when creating spooled files.
3. Lock contention can occur on output queues or spool database files, resulting in bottlenecks and severe performance problems.

You can reclaim empty spooled file members by:
- Adjusting the value assigned to the Reclaim Spool Storage (QRCLSPLSTG) system value.

- Using the Reclaim Spool Storage (RCLSPLSTG) command.

Example 1: Adjusting the QRCLSPLSTG system value

Let's assume that one of your application programs had an error and it produced thousands of spooled files that were of no value to you. When this happened those spooled files used lots of storage space on your system.

Change the QRCLSPLSTG system value to 1.

Delete all the unwanted spooled files that the application program created. Note the time of day you deleted all the unwanted spooled files.

After 24 hours, provided the empty spooled file members are not reused, the system reclaims the auxiliary storage that was being used by the empty spooled files.

Change the QRCLSPLSTG system value back to its former value.

Example 2: Using the reclaim spool storage (RCLSPLSTG) command

Let's assume that one of your application programs had an error and it produced thousands of spooled files that were of no value to you. When this happened those spooled files used lots of storage space on your system.

Delete all the unwanted spooled files that the application program created.

Run the RCLSPLSTG command with the DAYS parameter set to *NONE.

The system immediately reclaims all auxiliary spool storage that was being used by the unwanted spooled files.

Reducing system ASP storage

You can reduce the amount of storage taken up by spooled files by moving or creating spooled files directly into a user ASP. You can accomplish this by specifying SPLFASP(*OUTQASP) when creating an output queue in a library that is located in the desired user ASP.

All spooled files you place in this output queue will have the spooled file data stored in the user ASP in a library QSPLxxxx where xxxx is the user ASP number.

Note: The links to the job still reside on the system ASP. If the system ASP is lost, all spooled files, including those in the user ASPs, are lost. If a user ASP is lost, only spooled files in that user ASP are lost.

Spooled file names

When spooled files are created, the spooled file name is usually the same as the name of the printer file that was used to create it. For example, if the Print key is pressed the spooled file would be called QSYSPRT, because QSYSPRT is the printer file used by the Print key operation.

There are several ways in which the spooled file could have a different name:
- The Override with Printer File (OVRPRTF) command was used and a name was specified in the SPLFNAME parameter. For example, typing the following command:

```
OVRPRTF QSYSPRT SPLFNAME(REPORT1)
```

causes the name of the spooled file to be REPORT1 instead of QSYSPRT.

- The OVRPRTF command was used and a different printer file is specified in the TOFILE parameter. For example, typing the following command:

```
OVRPRTF QSYSPRT TOFILE(PRTF2)
```

causes the spooled file to be called PRTF2 (the name of the printer file specified in the TOFILE parameter of the OVRPRTF command).

- Some IBM applications may create spooled files that have names different from the printer files used to create them. Users have no control over spooled file names in this situation.

Redirecting spooled files

You may want to move some of your spooled files to an output queue that has no spooled files currently in it so that they will print sooner. To do this, enter the WRKSPLF command and type a 2 (Change) next to each spooled file you want to move. Press the Enter key and, in the OUTQ parameter, type the name of the output queue you want each spooled file to go to.

File redirection occurs when a spooled file is sent to an output device other than the one for which it was originally intended. File redirection may involve devices that process different media (such as printer output sent to a diskette device) or devices that process the same type of media but are of different device types (such as 5219 Printer output sent to a 4224 Printer).

Depending on the new output device for the spooled file, the file may be printed just as it would have been on the originally specified device. However, differences in devices often cause the output to be formatted differently. In these cases, the system sends an inquiry message to the message queue of the printer writer program to inform you of the situation and allow you to specify whether you want printing to continue.

Spooled files created by printing an OfficeVision/400™ document should not be redirected to a different output device than was originally specified in the Print Options of the document. The document will not print correctly.

For more information about spooled file redirection, see "Redirecting output" on page 112.

Copying spooled files

You can use the Copy Spooled File (CPYSPLF) command to copy a spooled file to a physical file.

Note: If the data stream type is *USERASCII, *AFPDS, *AFPDSLINE, or *LINE, (determined by the DEVTYPE parameter on the printer file) you cannot copy the spooled file.

The original spooled file is not affected by the copy operation and can still be printed by the printer writer program. You may want to copy a spooled file to a database file for the following reasons:

- You may want to copy the spooled file to a physical file because there are no commands to save spooled files on tape or diskette. However, if the system fails, physical files are backed up and you can recover the data. You can use the spool APIs to create your own save and restore operations for spooled files.

- You can save paper by having reports produced on microfiche instead of printed.
- You can copy a report to a database file to be sent over communications lines for printing at another location.
- You have collected information in a spooled file by running one of the system display commands with OUTPUT(*PRINT) specified. The spooled file created by this operation can be copied to a database file so that it can be read and processed by the application program using it.

  Note: If you use this method of gathering information, remember that the system displays that you are spooling may change when new functions are added to the system.

- You can copy the file to a spooled file so you can direct the same output to a different output queue.

  When copying spooled files to a database file, many device requirement attributes of the spooled file cannot be copied. Most OfficeVision/400 documents have device requirements such as variable lines per inch (LPI) and characters per inch (CPI) that are not copied to a database file. If the database file is copied back to the spooled file, the spooled file will not print the same as the original file.

Selecting the control character (CTLCHAR) parameter

The CTLCHAR parameter determines which control code is produced by the Copy Spooled File (CPYSPLF) command. You can select one of the following control codes:

- *NONE: No print control characters are created. You can use this code, for example, when printed displays (produced with the OUTPUT(*PRINT) parameter) are to be read by an application program.
- *FCFC: The first character of every record contains one of the following American National Standard control codes:

Code	Action before Printing a Line
' '	Space one line (blank code)
0	Space two lines
–	Space three lines
+	Suppress space
1	Skip to next channel 1
2	Skip to next channel 2
3	Skip to next channel 3
4	Skip to next channel 4
5	Skip to next channel 5
6	Skip to next channel 6
7	Skip to next channel 7
8	Skip to next channel 8
9	Skip to next channel 9
A	Skip to next channel 10
B	Skip to next channel 11

C   Skip to next channel 12

You can use the first-character forms-control code to create microfiche with the database file or print the spooled file with the Copy File (CPYF) command (which allows spooled files to be printed using a first-character forms-control printer file).

- *PRTCTL: The first 4 characters of every record contain skip-before and space-before values. This code can be viewed as `sssl` where `sss` is the skip-before line value (001 to 255) and `l` is the space-before value (0, 1, 2, or 3). When one part of the code is created by the CPYSPLF command, the other part is blank. In the following examples, ƀ represents a blank:

    '005ƀ' Skip to line 5 before printing

    '099ƀ' Skip to line 99 before printing

    'ƀƀƀ1' Space 1 line before printing

    'ƀƀƀ0' Do not space (or skip) before printing

  You can use this control code when printing with an RPG/400 program if the page size is not more than 99 lines long. To do this, move the control characters into the RPG/400 program's PRTCTL data structure space before and skip before fields and then print line.

  Note: Any skip to line values of 3 or less will actually be generated as space 1 line(s) to get to the correct line. For example; skip to line 2 would generate a space 1 line to get to line 2.

- *S36FMT: Specifies that the format of the records copied to a database file is the same as created by $UASF on System/36™ for COPYPRT. Only spooled print files can be copied when *S36FMT is specified. You can use this option when you plan to send the spooled file to a System/36. You should copy to a database file that has a record length of 150, 215, or 248. These are the valid record lengths for a System/36 file.

  The first record placed in the database file for each spooled file to be copied is a heading record. Columns that are not defined are blank.

Table 9 and 10 describe the formats of the header and data records.

Table 9. Header record format

Beginning Column	Field Length	Description
1	1	The letter H (to indicate the heading record).
4	6	The spool ID of the entry. Valid spool IDs range from SP0000 to SP9999, and from A00000 to Z99999.
12	8	The procedure name. This is blank if the file was not created by a procedure.
22	8	The job name. (The last 2 characters of the name are truncated.)
32	8	The user ID of the spooled file creator. (The last 2 characters of the ID are truncated.)
42	8	The printer device file name. (The last 2 characters of the name are truncated.)
52	2	The System/36 printer ID that corresponds to the device the file is printed on. The printer ID shown is the ID for the System/36 operating environment that copies the file, not the environment that created the spooled file.
56	4	The forms identification. (This is the first 4 characters of the form type of the spooled file.)

Table 9. Header record format (continued)

Beginning Column	Field Length	Description
61	2	The number of copies (in binary).
65	2	The number of pages (in binary).
69	4	The number of records (in binary). This is the number of data records that follow this heading record.
74	2	The number of lines per page (in binary).
78	1	The letter I if this entry contains print records with double-byte character set data.
81	1	The letter M if this entry contains print records with a length greater than 132.
84	1	Lines per inch (in binary).
85	1	Characters per inch (in binary).
86	1	Font ID (in binary). The printer file FONT parameter is converted to a binary 1 field. The maximum font ID on System/36 OCL is 255; the iSeries server supports font IDs above 255. Any time a font ID above 255 is used, this field is set to X'0B' for Courier 11 font. If you specify FONT(*CPI), the field contains X'00'.
87	1	Justify. Valid values are X'00' (0%), X'32' (50%), and X'64' (100%).
88	1	Align. (Y means to align forms, N means not to align forms.)
89	2	The maximum length of the print lines in the spooled file that was copied.
92	10	The nontruncated user ID of the spooled file creator.
102	10	The nontruncated printer file name.
112	10	The nontruncated form type.
113	7	(Used internally by the system.)

The data records placed in the disk file for each copied spooled file have the following format:

Table 10. Data record format

Beginning Column	Field Length	Description
1	2	The page number (in binary).
3	2	The line number (in binary).
5	4	The record number (in binary).
9	1	The letter I if this print record contains double-byte character set data.
10	1	A double-byte character set shift-out character (hex 0E) if this print record starts with double-byte character set data.
11	nnn	The data to be printed. (The field length is the file record length minus 10. If the print data is longer than the field length, it is truncated; and if it is shorter than the field length, it is padded with blanks at the end.) The format of data may not exactly match the format that would be produced on System/36 when multiple prints are used to construct a single print line.

The 2-byte binary numbers are unsigned, which means that a page number of 65 535 is the largest page number in a heading record or data record. When the actual number is larger, it will wrap beyond 65 535 to 0, then 1, 2, 3, and so on.

Selecting the channel value (CHLVAL) parameter

You can use the CHLVAL parameter on the CPYSPLF command to assign line numbers for the different channels described above. This can only be specified if *FCFC is specified for the CTLCHAR parameter. The same channel values that were specified when the file was originally spooled should be specified when using the CPYSPLF command.

Example of using control codes

To copy a spooled file named ORDERS in job NEWORDERS to database file PRTORDERS so that you can use the CPYF command to print the database file, type:

```
CPYSPLF   FILE(ORDERS) JOB(NEWORDERS) TOFILE(PRTORDERS) CTLCHAR(*FCFC)
```

To print the data from the physical file, type:

```
OVRPRTF   FILE(QSYSPRT) CTLCHAR(*FCFC) CHLVAL(*NORMAL)
CPYF      FROMFILE(PRTORDERS) TOFILE(QSYSPRT)
DLTOVR    FILE(QSYSPRT)
```

Sample commands for additional spooling support

You can define some functions to provide additional spooling support. Example source and documentation for the commands, files, and programs for these functions are part of library QUSRTOOL, which is an optionally installed part of the OS/400 program.

Documentation about these commands can be found in library QUSRTOOL, file QATTINFO, and the following members:

Member Name	Name of Command
TSRINFO	Save Spooled File command
TSRINFO	Restore Spooled File command

Descriptions of spooling commands in QUSRTOOL are:

Save spooled file (ZSAVSPLF) sample command

This command allows you to save a spooled file or a group of spooled files into a library and optionally save the library to a device.

Restore spooled file (ZRSTSPLF) sample command

This command allows you to restore spooled files from a library and optionally restore the library list from a device. The ZRSTSPLF command works only with spooled files saved using the ZSAVSPLF command.

Working with job and file separators

Separator pages are pages that separate one print job or spooled file from another. Separator pages for print jobs (job separators) are specified on the output queue. For each job having spooled files on the output queue, the specified number of separator pages will print at the beginning of the printed output for each job. Separator pages for spooled files (file separators) are specified on the printer file or printer writer. The specified number of separator pages will print before each spooled file.

To print customized separator pages, for example, a banner separator page, specify a user exit program on the separator exit program (SEPPGM) parameter of the printer device description.

A sample exit program (in C and RPG) is in the QUSRTOOL library. See member TBSINFO in file QATTINFO in library QUSRTOOL for information about this sample exit program.

Using application program interfaces (APIs)

There are several APIs that provide access to spooled files. Using these APIs, a program can copy a spooled file to a database without any loss of information. See the OS/400 APIs in the iSeries Information Center for additional information about spool APIs.

Working with a QPRTJOB

A QPRTJOB job is a job that spooled files are associated with when the current job's user name is not the same as the user profile currently running. System jobs can change to run under a user's profile in order for a user to obtain ownership of the spooled file instead of the system job. For example, if you send a spooled file using the SNDNETSPLF command to user TINA on a different iSeries server, the file is spooled for job 999999/TINA/QPRTJOB. Spooling the file for this user's job instead of the system job makes sure that user TINA owns the spooled file. Then, when she runs the WRKSPLF command, the spooled file sent to her is shown.

Note: Using SPLFOWN parameter allows you to specify who owns the spooled file.

QPRTJOB jobs are created automatically by the system. There may be more than one QPRTJOB per user on a system. A QPRTJOB has a default value of 9999 spooled files. That number can be expanded to the maximum of 999,999 by changing the number in the QMAXSPLF system value. For more information about the QMAXSPLF system value, see the Work Management topic in the iSeries Information Center. When a user's QPRTJOB gets full, the system automatically creates a new one for the user. A separate QPRTJOB is created for each user that receives spooled files sent by the SNDNETSPLF command. If you use the SNDNETSPLF command to send users TINA and KEVIN spooled files, there would be jobs named 999999/KEVIN/QPRTJOB and 999999/TINA/QPRTJOB on the receiving system.

QPRTJOB jobs are created and used by a variety of system functions. For example:
- Using the SNDTCPSPLF or SNDNETSPLF commands to send a spooled file to another user on a different iSeries server.
- Sending a spooled file from VM or MVS™ through a VM/MVS bridge to an iSeries server.
- Receiving a spooled file using TCP/IP or the line printer daemon (LPD) process.
- Using the QSPCRTSP Spool API to create a spooled file for another user.
- Using the QWTSETP Security API to set the user profile to a different user and then create a new spooled file.

  Other applications that are running can use the QSPCRTSP and QWTSETP APIs resulting in additional QPRTJOB jobs on the system.
- Using the SETGID API, you can create a spooled file for a different, current, or group user profile when SPLFOWN is set to *CURGRPPRF.
- Using the SETUID API, you can set the user profile to a different user and then create a new spooled file for that user.

Cleanup of QPRTJOB jobs

QPRTJOB jobs continue to be reused until they have been inactive more than 24 hours. Inactive means all spooled files for the job have been deleted and no new ones have been received for that user in more than 24 hours. The cleanup is done by the system job QSPLMAINT.

Chapter 4. Remote System Printing

Remote system printing allows spooled files created on an iSeries server to be automatically sent to and printed on other systems.

The spooled output files are sent, from an output queue, using the Start Remote Writer (STRRMTWTR) command. The STRRMTWTR command allows spooled output files to be automatically sent to other systems using SNA distribution services (SNADS) or Transmission Control Protocol/Internet Protocol (TCP/IP).

Information provided is:
* Definition of the remote system printing function
* Benefits of using this function
* User print information
* Spooled file attributes associated with remote system printing
* Status of spooled files being sent to remote systems
* Examples of how to use remote system printing between iSeries servers, iSeries server to VM or MVS, and iSeries server to PS/2.

Benefits of Using Remote System Printing

* Output queue placement.

  Spooled files can be placed automatically on a specific output queue of a target system. This support is provided through the Create Output Queue (CRTOUTQ) and Start Remote Writer (STRRMTWTR) commands.
* Multiple remote writers increase throughput.

  Output queues can have multiple remote writers started to them. This allows multiple jobs to send spooled files simultaneously from one output queue.

  Note: 10 remote writers can be started to one output queue.
* One-command interface

  Once the environment (hardware and software) has been established, the Start Remote Writer (STRRMTWTR) command initiates all activity necessary to send spooled files to a remote system. An auto-start job entry exists in the QSPL subsystem that starts a job automatically when the QSPL subsystem is started. This job runs the STRRMTWTR command with the OUTQ parameter value set to *ALL. Therefore, a remote writer(s) is started to all output queues that have a remote system and a number of writers to autostart specified. Remote writers are also started to a remote output queue when the remote output queue is changed or a new one created.
* Distributed print routing with spooled file attributes

  For distributed print routing, spooled file attributes are available. They are:
  – User who created file

    This attribute identifies the user who created the spooled file.
  – System where file created

    This attribute identifies the system on which the spooled file was created.
  – User print information

This attribute is composed of the characters retrieved from the user-defined text.

Once a spooled output file is created with user defined text, the text cannot be changed. When the spooled output file is sent with the data format parameter value of *ALLDATA, the user print information becomes an attribute of the spooled file.

See "Working With User Print Information" on page 173 for more detail on how to work with the display, retrieve, and change user print information commands.

- Send (SND) and Defer (DFR) status' for spooled output files

  These status' enable you to monitor the activity of spooled output files.

  – SND

    The spooled output file is being sent or has been sent to a remote system

  – DFR

    The spooled output file has been deferred from being sent

  After spooled output files are successfully sent to a remote system (as best as can be determined), they are deleted or saved, as specified by the save spooled file attribute.

See "Send and Defer Status of Spooled Output Files" on page 174 for more detail on the send and defer attributes of a spooled file.

How remote system printing works

The diagram below illustrates the remote system printing function. The key commands in working with remote system printing are:

- Create Output Queue (CRTOUTQ)
- Start Remote Writer (STRRMTWTR)

An output queue is created to hold spooled output files. A remote output queue is an output queue created for use in sending spooled output files to a remote system. Several parameters on the CRTOUTQ command are required to do this. When these parameters are given values, we then have a *remote output queue*. The spooled output files on a remote output queue are sent by the remote writer(s) started to the output queue. Remote writers are automatically started based on the value specified on number of writers to autostart (AUTOSTRWTR) parameter. Or they can be started using the STRRMTWTR command.

The STRRMTWTR command starts the writer that sends the spooled output files on the remote output queue, to the remote system. The writer, which is a system job, takes spooled output files from a remote output queue and sends them to a remote system using SNADS or TCP/IP. The spooled output file can be sent to the same user who owns the spooled output file being sent, a specific output queue, or the output queue for the system printer on the target system. If the sending user profile doesn't exist on the target system, QNETSPLF user profile is used when using SNADS.

Note: When spooled output files are sent to an AS/400 system at Version 2 Release 3 or less (or a system with the destination type of *OTHER and using SNADS), the user profile to whom the spooled output files are being

sent must exist or be created on that target system.
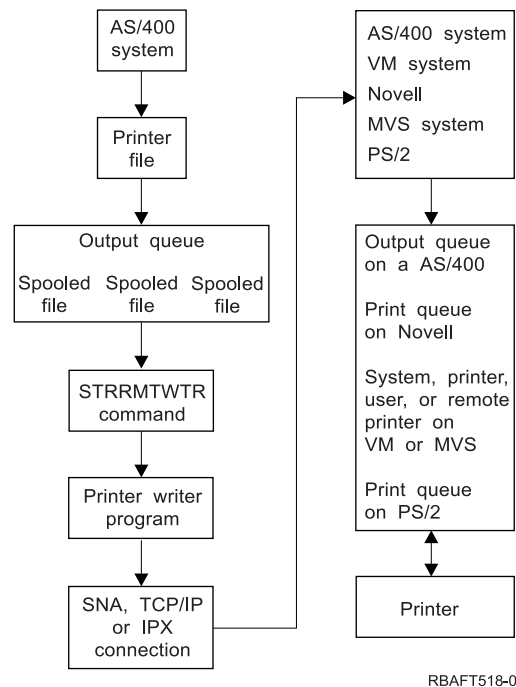
RBAFT518-0

The Role of the Create Output Queue (CRTOUTQ) Command in Remote System Printing

The CRTOUTQ command has certain parameters that allow the Start Remote Writer (STRRMTWTR) command to send spooled output files to remote systems.

The following parameters are important when working with remote system printing. However, not all the parameters of the CRTOUTQ command are listed here.

OUTQ

> The qualified name of the output queue.

> The name of the output queue can be qualified by one of the following library values:

> - *CURLIB
> - Specific library name

> The possible values are:

> **output queue name**
> > The name of the output queue being created.

MAXPAGES

> The maximum spooled file size in pages that is allowed to print between a starting and ending time. If a spooled file exceeds the page limit it is deferred (DFR status) until the ending time expires. For spooled files where the exact number of pages is not known, the estimated number of pages (DSPSPLFA) is used.

> **Notes:**

> 1. You can specify up to 5 combinations of pages and starting and ending times on the CRTOUTQ command.

2. The method of time measurement used is the 24 hour clock. For example, 2:00 PM would be entered as 1400.

Possible values are:

*NONE
> There is no limit on the size of spooled files allowed to print from this output queue.

Element 1: Number of Pages

number-of-pages
> The largest spooled file, in pages, that is allowed to print.

Element 2: Starting Time

starting-time
> The time of day that the maximum spooled file size limit is to start.

Element 3: Ending Time

ending-time
> The time of day that the maximum spooled output file size limit is to end.

RMTSYS
> The remote system to send spooled files to when a remote writer (STRRMTWTR command) is started to the output queue. This is referred to as the 'address' by SNADS, and the 'host' by TCP/IP. The possible values are:

*NONE
> The output queue is used only for local printing. The STRRMTWTR command cannot be used when this output queue is specified for the OUTQ parameter.

*PASTHR
> The system a user passed through from (STRPASTHR command) is used when sending spooled files created by the user job. If a spooled file was not created by a job that had passed through from another system, the spooled file will be held (HLD

*INTNETADR
> The INTNETADR parameter is used to identify the system when a remote writer is started to the output queue. This value is valid only when *IP has been specified for the CNNTYPE parameter status).

*NWSA
> The RMTPRTQ parameter is used to identify the system when a remote writer is started to the output queue. This value is valid only when *IP has been specified on the Connection type (CNNTYPE) parameter.

remote system name
> The name of the remote system. Only the first 8 characters are used when the connection type (CNNTYPE) parameter is specified as *SNA.

RMTPRTQ
> The printer queue on the remote system to which the remote writer sends spooled files. This RMTSYS parameter identifies the remote system.

*USER
> The user profile that creates a spooled file on the source system is also

the user ID that the spooled file is sent to on the target system. This value is valid only when the CNNTYPE parameter is *SNA.

*SYSTEM

The default system printer on the remote system determines the printer queue. This value is valid only when the CNNTYPE parameter is *SNA.

printer queue name

The name for the printer queue on the remote system. For remote systems that are iSeries servers, this is the name of an output queue that the spooled file is created on. For remote systems that are not iSeries servers, this name is system-dependent. The name can be either the actual name of the device or the name of a printer queue.

For iSeries servers, the output queue is usually specified as library name/output queue name. When the CNNTYPE parameter value is *SNA and the library name qualifier is not specified, the library list of the user on the target system is searched for the specified output queue. When the CNNTYPE parameter value is *IP, the QUSRSYS library is searched

The following table identifies the remote output queue on the target system that receives the spooled output files depending on the value of the RMTPRTQ parameter.

Table 11. Remote Printer Queue (RMTPRTQ) Values and Destination Type (DESTTYPE) Values

Destination Type (DESTTYPE)[1]	Remote Printer Queue (RMTPRTQ)		
	*USER	***SYSTEM**	**Printer queue name**
*OS400	File sent to user ID QNETSPLF. File then spooled to the default output queue of the original owner of the file. If the user profile of the original owner doesn't exist on the target system, the file is spooled to the default output queue for user profile QNETSPLF.	File sent to user ID QNETSPLF. File then spooled to the output queue of designated system printer (QPRTDEV). File is owned by a user profile with the same name as the original user profile owner of the spooled file. If this user profile is not found, the QNETSPLF user profile is used.	File sent to user ID QNETSPLF. File then spooled to the output queue specified by the printer queue name. File is owned by a user profile with the same name as the original user profile owner of the spooled file. If this user profile is not found, the QNETSPLF user profile is used.
*OS400V2	File sent to the file owners user ID on the target system. File spooled to that user ID's default output queue. If the sending user ID doesn't exist on the target system, an error is returned to the source system.	Not Allowed	The printer queue name is assumed to map to a user ID on the target system. The file is sent to this user ID and spooled to the user ID's default output queue.
*S390	File sent to the file owners user ID on the target system.	File sent to the system printer.	Sent to remote printer queue name specified

Table 11. Remote Printer Queue (RMTPRTQ) Values and Destination Type (DESTTYPE) Values (continued)

Destination Type (DESTTYPE)[1]	Remote Printer Queue (RMTPRTQ)		
	*USER	*SYSTEM	Printer queue name
Other (OS/400 V1, S/36, S/38)	File sent to the file owners user ID on the target system. File spooled to that user ID's default output queue. If the sending user ID doesn't exist on the target system, an error is returned to the source system.	Not Allowed	The printer queue name is assumed to map to a user ID on the target system. The file is sent to this user ID and spooled to the user ID's default output queue.
Note: [1] These connections, to the destination (or target system), are all through SNA.			

AUTOSTRWTR

The number of remote writers that are started automatically by the source system. For user-created output queues with the remote system (RMTSYS) parameter specified as *NONE, this parameter is ignored.

1 For system-created default printer output queues, this value is specified as 1 when the output queue is created.

*NONE

No writers are auto-started by the system to this output queue.

number of writers

The number of writers, between 1 and 10, to be auto-started to this output queue.

MSGQ

The qualified name of the message queue to which messages created by the remote writer started to this output queue are sent.

The name of the message queue can be qualified by one of the following library values:

- *LIBL
- *CURLIB
- Specific library name

Possible values are:

QSYSOPR

Messages are sent to the QSYSOPR message queue.

message queue name

The name of the message queue to which messages created by the remote writer are sent.

CNNTYPE

The type of connection with the remote system. This parameter determines the protocol used by the remote writer when sending spooled files. Possible values are:

*SNA Spooled files are sent using SNADS. This is similar to the Send Network Spooled File (SNDNETSPLF) command and requires that SNADS be configured.

*IP Spooled files are sent using TCP/IP. This is similar to the Send TCP/IP

Spooled File (SNDTCPSPLF) command and requires that the TCP/IP product is installed. TCP/IP must also be installed on the target system.

*USRDFN

The spooled files are sent by using a user-defined connection.

The following table summarizes the valid combinations between the connection type (CNNTYPE) and the destination type (DESTTYPE) parameters.

Table 12. Valid combinations of CNNTYPE and DESTTYPE

Destination Type (DESTTYPE)	Connection Type (CNNTYPE)		
	*SNA	***IP**	***USRDFN**
*OS400	X	X	X
*OS400V2	X	N/A	X
*PSF2	N/A	X	X
*S390	X	N/A	X
*NDS	N/A	X	X
*OTHER (OS/400V1, S36, S38)	X	X	X

DESTTYPE

The type of the remote system (RMTSYS) parameter. This parameter, along with the type of data contained in the spooled file (DEVTYPE parameter on the CRTPRTF command), is used by a remote writer to determine the format used to send the spooled file.

The spooled file is held by the remote writer if the type of data in the spooled file is not supported by the system. Possible values are:

*OS400

When the connection type (CNNTYPE) has been specified as *SNA, spooled files are to be sent to an iSeries server or AS/400 system running OS/400 version 3 or later. *OS400 can be specified for all releases which support the SNDTCPSPLF command (version 2 release 3 and later) when CNNTYPE is *IP or *USRDFN.

Note: This value should be specified when possible. It allows the greatest flexibility when selecting values for other parameters.

*OS400V2

Spooled files are to be sent to an AS/400 system running OS/400 version 2. This value is only valid when CNNTYPE is *SNA or *USRDFN.

*S390 Spooled files are to be sent to S/390® system. This value is only valid when CNNTYPE is *SNA or *USRDFN.

*PSF2 The spooled files are to be sent to a personal computer running the PSF for OS/2 product. This value is valid only when the CNNTYPE is *IP or *USRDFN.

*NDS The spooled files are sent to NETWARE4. This value is only valid when the CNNTYPE is *IP or *USRDFN.

*OTHER

Spooled files are to be sent to a system not matching any of the other

special values. This includes AS/400 systems running OS/400 version 1, as well as System/36 and System/38™.

CLASS
The VM/MVS SYSOUT class for spooled files sent to a VM or MVS host system.

Note: This parameter is valid only when CNNTYPE(*SNA) and DESTTYPE(*S390) are specified.

A The class is A.

class value
Distribution class value. Valid values range from B through Z and 0 through 9.

FCB
The FCB (forms control buffer) for spooled output files sent to a VM or MVS host system.

Note: This parameter is valid only when CNNTYPE(*SNA) and DESTTYPE(*S390) are specified.

*NONE
No FCB is used when sending files.

*USRDTA
The first 8 characters of the user data (USRDATA) spooled file attribute determines the name of the FCB. If the user data is blank, no FCB is used.

*PRTF The first 8 characters of the printer file used to spool the spooled file determines the name of the FCB.

FCB name
The name of the forms control buffer to be used.

TRANSFORM
Whether or not to use the host print transform function to transform a spooled file of device type *SCS or *AFPDS into ASCII data when the spooled file is sent by a remote writer.

Note: This parameter is valid only when the CNNTYPE parameter is *IP or *USRDFN.

Possible values are:

*YES Transform SCS or AFPDS data streams into ASCII.

*NO Do not transform SCS or AFPDS data streams in ASCII.

USRDTATFM
Specifies the qualified name of a user data transform program that is used to transform the spooled file data. The possible values are:

*NONE
No user-defined data transform program is specified.

name Specifies the name of the transform program. It can be qualified by one of the following library values:

*LIBL All libraries in the job's library list are searched until the first match is found.

*CURLIB
> The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

Library-name
> Specifies the library to be searched.

Note: The user data transform program is only valid when the remote system value is not *NONE. This parameter is not valid when TRANSFORM(*YES) is specified.

MFRTYPMDL
The manufacturer, type, and model for a printer using the host print transform function.

This parameter is only valid when TRANSFORM(*YES) or a user data transform program is specified. The possible values are:

*IBM42011
> The IBM 4201-1 Proprinter* is used.

*WSCST
> The value of the WSCST parameter is used.

manufacturer-type-model
> The manufacturer, type, and model for a printer using the host print transform function.

> **Note:** Prompting (F4) for this parameter shows many printers that can be used as a value for this parameter.

WSCST
An object that consists of a table of attributes used to customize a given ASCII device, such as a workstation or printer. Character presentation, font specifications, and control key sequences are examples of characteristics that can be customized.

This parameter is only valid when TRANSFORM(*YES) or a user data transform program is specified. Possible values are:

*NONE
> Do not specify a workstation customizing object.

> The name of the workstation customizing object can be qualified by one of the following library values:
> - *LIBL
> - *CURLIB
> - Specific library name

workstation-customizing-object
> A valid workstation customizing object which has been created with the Create Work Station Customizing Object (CRTWSCST) command.

IMGCFG
The IMGCFG parameter is used to enable the image print transform function. For more information about image configurations. See Chapter 14, "Working with the image print transform function" on page 291.

DESTOPT
Specifies destination-dependent options.

Specifying CNNTYPE(*IP) adds the options that are listed below to the control file that is sent to the LPD server. Specifying CNNTYPE(*IP) and DESTTYPE(*NDS) indicates whether OS/400 should track the spooled file on the destination system's printer queue until processing is complete. For CNNTYPE (*SNA), DESTTYPE (*OS400), or DESTTYPE (*OS400V2), the destination option indicates whether OS/400 should wait for confirmation from the destination system. The possible options and their values are:

*NONE
>No destination option specification.

*USRDFNTXT
>The user-defined text of the user profile at the spooled file creation. When you specify CNNTYPE(*IP) or CNNTYPE(*SNA), this value is ignored.

*NOWAIT
>When specifying CNNTYPE(*IP) or CNNTYPE(*SNA), once the spooled files are sent, OS/400 will no longer keep track of them.

Destination options specifiy destination-dependent options.

INTNETADR
>The internet address of the remote system to which the print request is sent.
>
>This parameter is only valid when *INTNETADR is the value for the RMTSYS parameter and *IP or *USRDFN is the value for the CNNTYPE parameter. Possible values are:
>
>**internet-address**
>>The internet address is specified in the form *nnn.nnn.nnn.nnn*, where *nnn* is a decimal number ranging from 0 through 255. (An internet address having all binary ones or zeros in the bits of the network or host identifier portions of the address is not valid.)

USRDFNOPT
>Specifies one or more user-defined options to be used by user applications or user-specified programs that process spooled files. The possible values are:
>
>***NONE**
>>No user-defined options are specified.
>
>**user-defined-option**
>>User-defined option to be used by user applications or user-specified programs that process spooled files.

USRDFNOBJ
>Specifies a name or type of the user-defined object to be used by user applications or user-specified programs that process spooled files. The possible values are:
>
>***NONE**
>>No user-defined objects are specified.
>
>**Element 1: Name of user-defined object**
>>The name of the user-defined object can be qualified by one of the following library values:
>>
>>***LIBL** All libraries in the job's library list are searched until the first match is found.

*CURLIB
> The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

Library-name
> Specifies the library to be searched.

Object name
> User-defined object to be used by user applications or user-specified programs that process spooled files.

Element 2: Type of user-defined object
> The user object type can be one of the following:
> - *DTAARA - Data Area
> - *DTAQ - Data Queue
> - *FILE - File
> - *PSFCFG - PSF Configuration Object
> - *USRIDX - User index
> - *USRQ - User Queue
> - *USRSPC - User Space

USRDRVPGM

Specifies the name of the user-specified driver program. The possible values are:

*NONE
> No driver program is specified.

name   Specifies the name of the user driver program. The name of the driver program can be qualified by one of the following library values:

> ***LIBL**   All libraries in the job's library list are searched until the first match is found.

> ***CURLIB**
>> The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used

> **library-name**
>> Specifies the library to be searched.

> **User driver program name**
>> User-specified driver program to process spooled files.

>> **Note:** The user driver program is only valid when the remote system value is not *NONE.

Role of the Start Remote Writer (STRRMTWTR) Command When Using Remote System Printing

The Start Remote Writer (STRRMTWTR) command starts a spooling writer that sends spooled output files from an output queue on a source system to a remote system (target system).

Note: When the spooled output file is composed of an AFPDS data stream that uses AFP resources (for example, page segments or overlays), those resources must exist on the target system if they are to be included in the

printed output. OS/400 does not insert them into the spooled output file before sending the spooled output file to the target system. This applies only if you are not transforming the spooled file using the host print transform function.

More than one writer can be active at the same time (as determined by the spooling subsystem description). Up to 10 writers can be active to the same output queue. Each writer must have a unique writer name, and only one type (printer, remote, or diskette) of writer can be active to a single output queue. A writer that is started can be actively sending spooled files or waiting for a spooled file to be put on the output queue. Optionally, the writer can end automatically when it has processed all the spooled files on the output queue. You can also change, hold, or cancel the writer.

The following table shows the data format of the spooled file that is sent to the destination system when the connection type is *SNA.

*Table 13. Connection Type (CNNTYPE) *SNA*

Destination Type (CNNTYPE)	Data Format	
	***ALLDATA**	***RCDDTA**
*OS/400	All device type	
*OS/400 V2	All device type	
*OTHER (OS/400 V1, S/36, S/38)		All device type
*S390	*LINE, *AFPDSLINE, *AFPDS	*USRASCII, *IPDS, *SCS

Note: All device type includes *LINE, *AFPDSLINE, *AFPDS, *USRASCII, *IPDS, and *SCS.

You can continue with other work after starting a writer because each job runs independently. The writer is owned by the user who issues the STRRMTWTR command. The following are the parameters and values for the STRRMTWTR command:

OUTQ
The qualified name of the output queue. Possible values are:

*ALL   Remote writers are started for every output queue on the system which has a remote system (RMTSYS parameter value) specified.

The number of writers started to a particular output queue is determined by the value specified on the AUTOSTRWTR parameter of the CRTOUTQ command.

No writer is started if an output queue already has the number of writers to autostart started or a different type of writer (diskette or printer) is started to the output queue.

The name of the output queue can be qualified by one of the following library values:
- *LIBL
- *CURLIB
- Specific library name

Output queue name
The output queue name from which the printer writer processes spooled files.

output-queue-name
> The output queue from which the printer writer processes spooled files.

WTR
The spooling writer being started. Each writer name must be unique. Possible values are:

*OUTQ
> The name of the writer is the same as that of the output queue specified on the OUTQ parameter. If the output queue has been specified as *ALL and the number of writers to autostart is greater than 1, the writer name is generated by taking the first nine characters of the output queue name and appending a successive numeric character (2 through 0) to it.

*SYSGEN
> The writer name is generated by the system, starting with 'RMTW000001' and incrementing the numeric part for each successive writer.

Writer name
> The name by which the writer being started is identified.

MSGQ
The qualified name of the message queue to which messages created by the writer are sent. Possible values are:

*OUTQ
> Messages are sent to the message queue specified in the output queue of the output queue named on the OUTQ parameter.

*REQUESTER
> The messages are sent to the message queue of the user who started the writer. If this value is specified for a batch job, *OUTQ is used.
>
> The name of the message queue can be qualified by one of the following library values:
> - *LIBL
> - *CURLIB
> - Specific library name

Message queue name
> The message queue name to which messages created by the writer are sent.

AUTOEND
Specifies whether the writer ends automatically.

*NO The writer does not end when the last available spooled file has been removed from the output queue. It waits for another spooled file entry to be put on the queue.

> **Element 1: Ending a Writer Automatically**

*YES The writer automatically ends after it has reached the state specified on Element 2 of this parameter (*NORDYF or *FILEEND).

> **Element 2: Ending a Writer with a Ready File**

*NORDYF

> The writer automatically ends when there are no ready spooled files (all the available spooled files have been removed from the output queue).

*FILEEND

> The writer ends after it finishes processing one spooled file.

FORMTYPE

The name of the form type. The writer uses this value to select spooled files from the specified output queue for sending to a remote system.

Note: A spooled file's form type is specified in the device file that produced the spooled file.

This is important because the output queue can contain spooled files with many different form types. Only the spooled files with the specified form type are sent; others wait on the output queue.

Communication with the remote system is important to determine which form types their system supports. When the form type is known, the spooled files can be created with the desired form type. You could also use the Change Spooled File Attributes (CHGSPLFA) command. However, this is not as effective as creating the spooled files with the needed form type.

Element 1: Type of Form Designation

Possible values are:

*ALL All form types are processed by the writer.

*FORMS

> Available spooled files on the output queue with the same form type are processed as a group before the writer moves on to the next form type group. The writer first chooses the first available spooled file on the output queue.
>
> After the first spooled file is complete, all spooled files with the same form type are processed. The writer again chooses the first available spooled file on the output queue and repeats the process for that form type.

*STD Only spooled files that specify the standard form type are selected.

Form type

> The form type of the spooled files being produced.

Element 2: Message Sending Options

Note: The message sending options are most effective when the source system is able to send the spooled files directly to a printer queue on the target system.

*NOMSG

> Neither an inquiry message nor an informational message is sent to the message queue.

*INQMSG

> An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type last sent.

*INFOMSG
> An informational message is sent to the message queue when no spooled files requiring this form type remain in the output queue.

*MSG An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type in the remote and an informational message is sent when no spooled files requiring this form type remain in the output queue.

Working With User Print Information

User print information is made up of user-defined text associated with a user. The user-defined text is saved with the spooled file when the spooled file is created. It can be displayed using the Display Spooled File Attributes (DSPSPLFA) command or retrieved using the Retrieve User Print Information (RTVUSRPRTI) command.

User print information is not used when sending spooled output files to another iSeries server or an S/3X system. It is used only as information to pass to the VM/MVS bridge customer user exit to aid in setting the Network Job Entry (NJE) header fields.

It's possible for a system administrator to limit users access by revoking public authority to the particular commands.

User print information can be used any way you want. For example, it could consist of printout distribution information or be used for accounting information (department to charge for printing).

User print information is worked with using the Change User Print Information (CHGUSRPRTI), Display User Print Information (DSPUSRPRTI), and Retrieve User Print Information (RTVUSRPRTI) commands.

Using the CHGUSRPRTI Command

There is no command that allows creation of user print information. If no user print information exists, it can be created using the CHGUSRPRTI command.

For example, running the following command modifies (or creates if it doesn't exist) the user print information for user LAWSON.

```
CHGUSRPRTI USER(LAWSON)  TEXT('DEPT. 269 P.O. BOX 123')
```

The command acts on the user print information for the user LAWSON. The user information is changed (or created) to DEPT. 269 P.O.Box 123.

Using the DSPUSRPRTI Command

The Display User Print Information (DSPUSRPRTI) command displays the user print information for the specified user.

```
DSPUSRPRTI USER(LAWSON)
```

Note: DSPUSRPRTI uses the QPDSPUSRPI printer file when OUTPUT is specified as *PRINT.

Using the RTVUSRPRTI Command

The Retrieve User Print Information (RTVUSRPRTI) command can be used in a CL program to retrieve the user print information value associated with a user. The values are returned in the specified CL variables for that user.

```
RTVUSRPRTI USER(LAWSON) RTNTEXT(&TEXT);
```

When the above command is run, the following is returned:
```
&TEXT    'DEPT 269  P.O. BOX 123 ____'
```

The coded character set identifier (CCSID) is used when the text description prints on the output.

Send and Defer Status of Spooled Output Files

When a spooled output file is on an output queue, its status can vary depending on the mode of the output queue and/or the activity taking place with a particular spooled file.

Of particular interest to remote system printing are the status' of SND and DFR.

Note: DFR status is not unique to spooled output files on a remote output queue. Spooled output files on non-remote output queues can also have a status of DFR.

- SND

  When a spooled output file is being sent to a remote system, it has a status of SND. If the connection type is *SNA, the spooled output file may remain in SND status until a confirmation message from the remote system is received by the remote writer. At this time the spooled output file is deleted or saved, depending on the save spooled file attribute. If the writer is ended while spooled output files are in SND status, the spooled files are changed back to RDY status.

- DFR

  When a writer (printer or remote) is started to an output queue, it determines the maximum spooled file size for the current time. Any RDY spooled files which exceed the limit are changed to DFR status. If the spooled file exceeds the current limit and is added to the output queue (created or moved) after a writer is started to the output queue, the spooled output file status will be DFR.

  When the system time-of-day changes such that a new maximum spooled output file size is to take affect, the writer goes through the output queue again and updates RDY spooled files to DFR, or DFR to RDY, depending on the new limit and the size of the particular spooled output file. When the writer is ended, all DFR spooled output files return to RDY.

  When time ranges for the maximum spooled output file size overlap, the smaller of the number of pages values is used. For example, assume that there were two time ranges of 8:00:00 to 16:00:00 and 12:00:00 to 12:30:00, with number of pages 40 and 10 respectively. The largest spooled output file that would print from 8:00 am to 12:00 pm would be 40 pages. The largest spooled output file that would print from 12:00 pm to 12:30 pm would be 10 pages. The largest spooled output file that would print from 12:30 pm to 4:00 pm would be 40 pages.

The following screen illustrates an output queue (RMTOUTQ) with a status of released and the first spooled output file (DMB18R1) being written (RLS/WTR). Because DMB18R1 is being sent to a remote system, its status is SND. The next spooled output file, DMB18R2, has a status of DFR. It could be deferred from activity because of its size and the time of day that certain sizes of spooled output files are allowed to be printed or sent.

```
                    Work with Queue (WRKOUTQ *RMTOUTQ)

Queue:   RMTOUTQ          Library:   Lawson          Status:   RLS/WTR

Type options, press Enter.
1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
8=Attributes        9=Work with printing status

Opt  File        User       User Data   Sts   Pages   Copies  Form Type   Pty
 _   DMB18R1     LAWSON                  SND     1       1     *STD         5
 8   STUMPF      LAWSON                  RDY
 _   DMB18R2     LAWSON     TEST         DFR     1       1     *STD         5




                                                                     Bottom
Parameters for options 1, 2, 3 or command
===> _____
F3=Exit   F11=View 2   F12=Cancel   F20=Writers   F22=Printers
F24=More keys
```

Spooled Output File Attributes

Selecting option 8 from the Work with Output Queue screen displays all the
attributes of a spooled output file. The following three attributes are very useful in
determining origin and ownership of spooled output files.

- System where file created
- User who created file
- User print information

You can also display a spooled output file's attributes by using the Work with
Spooled Files (WRKSPLF) command and selecting option 8.

Preparing OS/400-to-OS/400 for Remote System Printing

The diagram below illustrates OS/400-to-OS/400 remote system printing with SNADS. The values (or names) in the diagram are used in this example.

```
Source system                                    Target system
Name: SYSNMD5T                                   Name: SYSNMDLC

STRRMTWTR to
output queue named                               Output queue
RMTOUTQ                                           on target system

                                                 OUTQ = OUTQRMT

Remote output queue                               Spooled file 1
named: RMTOUTQ                                    Spooled file 2
                                                  Spooled file 3
RMTSYS = SYSNMDLC                                 Spooled file 4
RMTPRTQ = OUTQRMT                                 Spooled file 5
  CNNTYPE = *SNA                                  Spooled file 6
 DESTTYPE = *OS400                                    .
                                                     .
                                                     .
                                                     .

   Spooled file 1
   Spooled file 2           STRPRTWTR command          STRPRTWTR command
   Spooled file 3           on target system           on target system
   Spooled file 4
       .                      DEV = PRT01                DEV = PRT02
       .                      OUTQ = OUTQRMT             OUTQ = OUTQRMT
       .
       .

                              Printer                    Printer
                              PRT01                      PRT02

                           Prints spooled files       Prints spooled files
                           from output queue          from output queue
                              OUTQRMT                    OUTQRMT
```

RBAFT519-0

Preparation work must be done on both the source system and target system for the remote system printing to work. Table 14 provides a list of things that must be present or created before remote system printing can be used.

Table 14. Source System and Target System Preparation for Remote System Printing

Source System	Target System
Line, controller, and device descriptions created Additional information for these configuration descriptions can be obtained in the Communications Configuration book.	Line, controller, and device descriptions created Additional information for these configuration descriptions can be obtained in the Communications Configuration book.
Identify the user profile(s) to be used for remote system printing. This information is needed if you want a corresponding user profile on the target system to own the sent spooled file(s). If you don't care who owns the spooled files, they can be sent to the target system and spooled to the QNETSPLF user profile.	Depending on where (to a user ID or output queue) you want the spooled files sent, you need to create a matching user profile or a specific output queue. Use the CRTUSRPRF or CRTOUTQ commands.

Source System	Target System

Setup the SNADS network.
 	
• Create distribution queues and routing entries using the Configure Distribution Services (CFGDSTSRV) command.
 	
  – Create a distribution queue.
 	
  – Create a routing entry and associate the target system name (SYSNMDLC) with the distribution queue.
 	
• Add users to the system directory. Use the Add Directory Entry (ADDDIRE) command.
 	
  – Add a directory entry for the QNETSPLF user profile. (This entry is used to send spooled files to the remote system.)
 	
    User ID
      This must be QNETSPLF.
 	
    Address
      This is the value specified on the RMTSYS parameter (SYSNMDLC).
 	
    User profile
      This must be *NONE.
 	
    System name
      This is the value specified on the RMTSYS parameter (SYSNMDLC).
 	
    System group
      Not used in this example.
 	
  – Add a directory entry for the QNETSPLF user profile. (This entry is the user ID used by SNADS on the source system.)
 	
    User ID
      This must be QNETSPLF.
 	
    Address
      This is the name of the source system (SYSNMD5T).
 	
    User profile
      This must be QNETSPLF.
 	
    System name
      This is the name of the source system (*LCL).
 	
    System group
      Not used in this example.
	Setup the SNADS network.
	
	• Create distribution queues and routing entries using the Configure Distribution Services (CFGDSTSRV) command.
	
	  – Create a distribution queue.
	
	  – Create a routing entry; associate the name of the distribution queue and the network ID of the target system (SYSNMD5T).
	
	• Add users to the system directory. Use the Add Directory Entry (ADDDIRE) command.
	
	  – Add a directory entry for QNETSPLF. (This entry is used to receive spooled files from the source system.)
	
	    User ID
	      This value must be QNETSPLF.
	
	    Address
	      This is the value specified on the RMTSYS parameter (SYSNMDLC).
	
	    User profile
	      This must be QNETSPLF.
	
	    System name
	      This must be *LCL.
	
	    System group
	      Not used in this example.
	
	  – Add another directory entry for user profile QNETSPLF. (This entry is used to send messages back to the source system).
	
	    User ID
	      This must be QNETSPLF.
	
	    Address
	      This is the name of the source system (SYSNMD5T)
	
	    User profile
	      This value must be *NONE.
	
	    System name
	      This is the name of the source system (SYSNMD5T)
	
	    System group
	      Not used in this example.

Create a remote output queue for remote system printing. Use the Create Output Queue (CRTOUTQ) command.	Create an output queue to receive the spooled files using the value on the RMTPRTQ parameter (OUTQRMT).

Source System Activity - Creating the Remote Output Queue

The result of using this example would be spooled output files sent to output queue OUTQRMT on the target system (SYSNMDLC).

Type CRTOUTQ and prompt (F4). The following screen appears. Type in the values shown and press the Enter key. The remote output queue RMTOUTQ is created. The other parameter values are needed to ensure connection and delivery to the designated system.

```
                    Create Output Queue (CRTOUTQ)

Type choices, press Enter.

Output queue . . . . . . . . . .   RMTOUTQ    Name
  Library  . . . . . . . . . . .    MYLIB     Name, *CURLIB
Maximum spooled file size:
  Number of pages  . . . . . . .    *NONE     Number, *NONE
  Starting time  . . . . . . . .              Time
  Ending time  . . . . . . . . .              Time
              + for more values
Order of files on queue  . . . .   *FIFO      *FIFO, *JOBNBR
Remote system  . . . . . . . . .   SYSNMDLC
Remote printer queue   . . . . .   OUTQRMT
Queue for writer messages  . . .   QSYSOPR    Name, QSYSOPR
  Library  . . . . . . . . . . .    *LIBL      Name, *LIBL, *CURLIB
Connection type  . . . . . . . .   *SNA       *SNA, *IP
Destination type . . . . . . . .   *OS400     *OS400, *OS400V2, *PSF2...
Text 'description' . . . . . . .   OS/400 to OS/400


                                                             Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

To begin sending spooled output files from the RMTOUTQ remote output queue, run the Start Remote Writer (STRRMTWTR) command. Type STRRMTWTR and prompt (F4).

The following screen appears. Type in the values for the parameters displayed and press the Enter key. Spooled output files are then sent to the OUTQRMT on the target system (SYSNMDLC).

```
              Start Remote Writer (STRRMTWTR)

Type choices, press Enter.

Output queue . . . . . . . . . .   RMTOUTQ    Name, *ALL
  Library  . . . . . . . . . . .    stumpf    Name, *LIBL, *CURLIB
Queue for writer messages  . . .   *OUTQ      Name, *OUTQ, *REQUESTER
  Library  . . . . . . . . . . .              Name, *LIBL, *CURLIB
Form type options:
  Form type  . . . . . . . . . .   *ALL       Form type, *ALL, *STD, *FORMS
  Message option . . . . . . . .   *NOMSG     *NOMSG, *INQMSG, *MSG...






                                                             Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Target System Activity - Printing Spooled Output Files from the Source System

The spooled output files were sent to the OUTQRMT output queue on the target system. To begin printing the spooled output files, start a printer writer to the OUTQRMT output queue. Type STRPRTWTR and prompt (F4).

Type in the name of the printer (CHEROKEE2) you want to use and the name of the output queue with the received spooled output files (OUTQRMT). Press the Enter key and spooled output files will begin to print on the printer named CHEROKEE2.

```
                   Start Printer Writer (STRPRTWTR)

 Type choices, press Enter.

 Printer  . . . . . . . . . . .   cherokee2  Name, *ALL, *SYSVAL
 Output queue . . . . . . . . .   OUTQRMT    Name, *DEV
   Library  . . . . . . . . . .    stumpf    Name, *LIBL, *CURLIB
 Queue for writer messages  . .   *DEVD      Name, *DEVD, *REQUESTER
   Library  . . . . . . . . . .              Name, *LIBL, *CURLIB
 Form type options:
   Form type  . . . . . . . . .   *ALL       Form type, *ALL, *STD, *FORMS
   Message option . . . . . . .   *INQMSG    *INQMSG, *MSG, *NOMSG...
 File separators  . . . . . . .   *FILE      0-9, *FILE
 Drawer for separators  . . . .   *DEVD      1-255, *DEVD, *FILE




                                                                Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display       F24=More keys
```

Multiple printer writers can be started to same output queue. This allows the spooled output files from a particular output queue to be printed on more than one printer. See "Using multiple printer writer support" on page 140 for more information on using multiple printer writers.

To see the number of writers started to the OUTQRMT output queue, run the WRKOUTQ *ALL command. The Work with All output queues screen appears. Type option 9 next to the OUTQRMT output queue and press the Enter key. The Work with All Writers display appears. If the output queue has more than one writer started to it, the writers are listed and they both are associated with the same output queue (OUTQRMT).

```
        Work with All output queues

Type options, press Enter.
2=Change   3=Hold     4=Delete   5=Work with   6=Release   8=Description
9=Work with Writers   14=Clear

Opt     Queue          Library        Files      Writer        Status
        SCCOUTQ        SCCLARK         156                      RLS
        T93            SCCLARK           0                      RLS
        AFP            SKS              23                      RLS
        AFP2           SKS               0                      RLS
        SKS2           SKS               0                      RLS
        DEFERQ         STANGLER          5                      HLD
        STANGLER       STANGLER         53                      RLS
        ANGELIKA       STUMPF            0                      RLS
  9     OUTQRMT        STUMPF            2      *CHEROKEE2       RLS
        RMTOUTQ1       STUMPF            0                      RLS
        TAAOUTQ        TAATOOL           0                      RLS
        TIEMENS        TIEMENS           0                      RLS
                                                                      More...
Command
===>
F3=Exit    F4=Prompt   F5=Refresh   F12=Cancel   F24=More keys
```

```
        Work with All Writers

Type options, press Enter.
2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display messages
8=Work with output queue

Opt  Writer    Type  Device    Queue     Library   Status  Form Type
     CHEROKEE2 PRT   CHEROKEE2 OUTQRMT   STUMPF     STR     *ALL
     CHIEF     PRT   CHIEF     OUTQRMT   STUMPF     STR     *ALL




                                                              Bottom
Parameters for options 2, 3, 4, 6 or command
===>
F3=Exit    F4=Prompt   F12=Cancel   F22=Start printer writer   F24=More keys
```

You can list the names of writers started to the OUTQRMT output queue by typing
WRKOUTQ OUTQRMT and prompting (F4). The Work with Output Queue
display appears. Press F20=Writers key and the Work with All Writers display
appears with a list of all writers that are started to the OUTQRMT output queue.
This sequence is shown in the following two displays:

```
                    Work with Output Queue

Queue:   OUTQRMT        Library:   STUMPF          Status:   RLS/WTR

Type options, press Enter.
1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
8=Attributes       9=Work with printing status

Opt  File       User       User Data   Sts   Pages   Copies  Form Type   Pty
     DMB18R2    XZZ0136    *BEFORE     HLD    4       1       ENTN        5
     QPRINT     STUMPF                 SAV    2       1       *STD        5




                                                              Bottom
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F11=View 2   F12=Cancel    F20=Writers   F22=Printers
F24=More keys
```

```
                    Work with All Writers

Type options, press Enter.
2=Change   3=Hold   4=End   5=Work with   6=Release   7=Display messages
8=Work with output queue

Opt  Writer      Type  Device      Queue       Library    Status  Form Type
     CHEROKEE2   PRT   CHEROKEE2   OUTQRMT     STUMPF      STR     *ALL
     CHIEF       PRT   CHIEF       OUTQRMT     STUMPF      STR     *ALL






                                                              Bottom
Parameters for options 2, 3, 4, 6 or command
===>
F3=Exit   F4=Prompt   F12=Cancel   F22=Start printer writer   F24=More keys
```

Preparing OS/400 to VM/MVS for Remote System Printing

The diagram below illustrates OS/400-to-VM/MVS remote system printing with SNADS. The values (or names) in the diagram are used in this example.

```
Source system
Name: SYSNMD5T

STRRMTWTR to
output queue
named RMTOUTQ
```

```
Target system
Name: Vm3

System printer
name defined by
VM system
```

```
System printer

FCB = FCB2
CLASS = C
```

```
Remote output queue
named: RMTOUTQ

    RMTSYS = VM3
  RMTPRTQ = *SYSTEM
   CNNTYPE = *SNA
  DESTTYPE = *S390
       CLASS = C
        FCB = FCB2

   Spooled file 1
   Spooled file 2
   Spooled file 3
   Spooled file 4
           .
           .
           .
           .
```
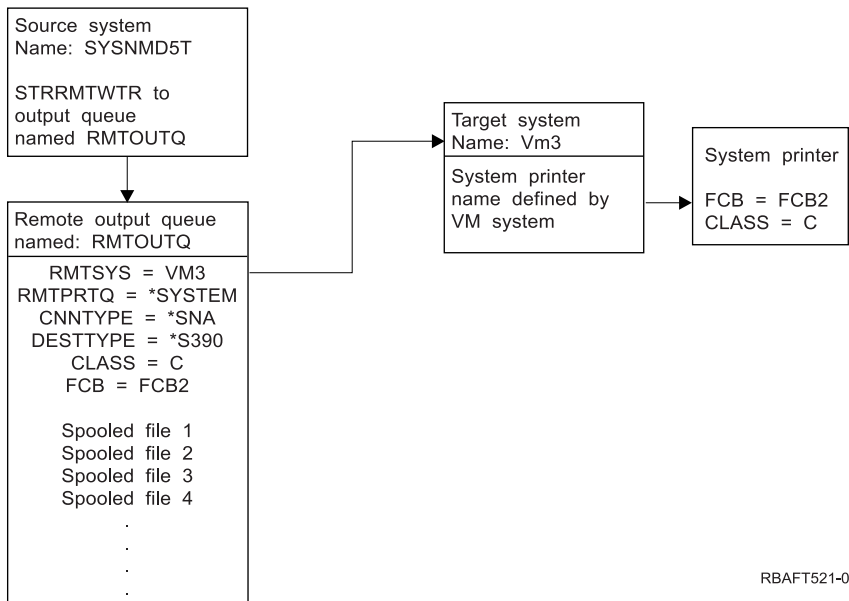
RBAFT521-0

Preparation work must be done on both the source system and target system for remote system printing to work. Table 15 provides a list of things that must be present or created before remote system printing can be used.

Table 15. Source System and Target System Preparation for Remote System Printing

Source System at V3R1 or later	VM/MVS Target System
Ensure Licensed Program 5769-CM1 is installed. This is needed for communications. For more details about configuring SNADS connections between these systems see the SNA Distribution Services book.	NJE configuration for VM/RSCS and JES2 or JES3 for MVS. Contact with the host system will be critical to establishing the correct connection with VM/MVS. System node name, printer ID, and user ID are some of the information OS/400 will require.
Line, controller, and device descriptions created Additional information for these configuration descriptions can be obtained in the Communications Configuration book.	Communication descriptions must exist or be created to connect the VM/MVS system to OS/400. Typically this would be a line, physical unit (similar to controller), and a logical unit (similar to device). The VM or MVS system configuration will require information from OS/400.
Identify the user profile to be used for remote system printing.	This user must also exist on the target system if *USER is specified on the RMTPRTQ parameter on OS/400.

Table 15. Source System and Target System Preparation for Remote System Printing (continued)

Source System at V3R1 or later	VM/MVS Target System

Setup the SNADS network.
• Create distribution queues and routing entries using the Configure Distribution Services (CFGDSTSRV) command.
  – Create a distribution queue.
  – Create a routing entry and associate the target system name (VM3) with the distribution queue.
• Add users to the system directory. Use the Add Directory Entry (ADDDIRE) command.
  – Add a directory entry for the QNETSPLF user profile. (This entry is the user ID used by SNADS on the source system.)
  User ID
    This must be QNETSPLF.
  Address
    This is the name of the source system (SYSNMD5T).
  User profile
    This must be QNETSPLF.
  System name
    This is the name of the source system (*LCL).
  System group
    Not used in this example.
  – Add a directory entry for QNETSPLF. (This entry is the user ID used by SNADS to route to the target system.)
  User ID
    This must be QNETSPLF.
  Address
    This is the name of the target system (VM3).
  User profile
    This must be *NONE.
  System name
    This is the name of the target system (VM3).
  System group
    Not used in this example.
	For MVS, identification of the iSeries server must be entered in the JCL or JES and associated with a printer defined to the MVS system.
	
	For VM, identification of the iSeries server must be entered in RSCS and associated with a printer defined to the VM system.

Create a remote output queue for remote system printing. Use the Create Output Queue (CRTOUTQ) command.	
When sending spooled output files to VM/MVS the default forms type must be STANDARD (not to be confused with *STD for OS/400). The forms type can be changed by overriding the printer file used to create the spooled output files. Also, you could use the Change Spooled File Attributes (CHGSPLFA) command and change the forms type parameter value to standard.	

Source System Activity - Creating the Remote Output Queue

The result of using this example would be spooled output files sent to the system printer on the target system (VM3).

Note: When a remote output queue is created, writers are automatically started to it if the AUTOSTRWTR parameter value is other than *NONE.

Type CRTOUTQ and prompt (F4). The following screen appears. Type in the values shown and press the Enter key. The remote output queue TEST1 is created. The other parameter values are needed to ensure connection and delivery to the designated system.

```
                    Create Output Queue (CRTOUTQ)

 Type choices, press Enter.

 Output queue . . . . . . . . . .   RMTOUTQ     Name
   Library  . . . . . . . . . . .     MYLIB     Name, *CURLIB
 Maximum spooled file size:
   Number of pages  . . . . . . .   *NONE       Number, *NONE
   Starting time  . . . . . . . .               Time
   Ending time  . . . . . . . . .               Time
               + for more values
 Order of files on queue  . . . .   *FIFO       *FIFO, *JOBNBR
 Remote system  . . . . . . . . .   VM3
 Remote printer queue   . . . . .   *SYSTEM
 Queue for writer messages  . . .   QSYSOPR     Name, QSYSOPR
   Library  . . . . . . . . . . .               Name, *LIBL, *CURLIB
 Connection type  . . . . . . . .   *SNA        *SNA, *IP
 Destination type . . . . . . . .   *S390       *OS400, *OS400V2, *PSF2...
 VM/MVS class . . . . . . . . . .   C
 FCB  . . . . . . . . . . . . . .   FCB2
 Text 'description' . . . . . . .   OS/400 to VM System named VM3
                                                                   Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display       F24=More keys
```

To begin sending spooled output files from the RMTOUTQ remote output queue, run the Start Remote Writer (STRRMTWTR) command by typing STRRMTWTR and prompting (F4).

The following screen appears. Type in RMTOUTQ and press the Enter key. Spooled output files are then sent to the printer designated as the system printer on the target system (VM3).

```
      Start Remote Writer (STRRMTWTR)

Type choices, press Enter.

Output queue . . . . . . . . . .   RMTOUTQ    Name, *ALL
  Library . . . . . . . . . . .     MYLIB     Name, *LIBL, *CURLIB
Queue for writer messages  . . .   *OUTQ      Name, *OUTQ, *REQUESTER
  Library . . . . . . . . . . .     *LIBL     Name, *LIBL, *CURLIB
Form type options:
  Form type . . . . . . . . . .    *ALL       Form type, *ALL, *STD, *FORMS
  Message option . . . . . . . .   *NOMSG     *NOMSG, *INQMSG, *MSG...




                                                            Bottom
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

Target System Activity - Printing Spooled Output Files from the Source System

The spooled files were sent to the system printer on the target system (VM3).

Preparing OS/400 to NetWare for Remote System Printing

See Printing from iSeries to NetWare printers in the Networking category of the iSeries Information Center for instructions on how you can print from iSeries to a NetWare printer that uses the standard NetWare print support and an iSeries remote output queue and remote writer.

Chapter 5. Working with the OS/400 Network Print Server

This chapter discusses the OS/400 network print server. The network print server is the host or print server for print clients.

The network print server provides client access to OS/400 print objects and resources. Following are the network print server objects and the actions that can be requested to be performed on these objects:

Objects

> **Actions**

Spooled file

> Create, seek, open, read, write, close, hold, release, delete, move, send, call exit program, change attributes, retrieve message, answer message, retrieve attributes, and list.

Writer job

> Start, end, and list

Printer device

> Retrieve attributes and list

Output queue

> Hold, release, purge, list, and retrieve attributes

Library

> List

Printer file

> Retrieve attributes, change attributes, and list

Network print server

> Change attributes and retrieve attributes.

How is the network print server accessed?

Before a client can communicate with the network print server, a communications session must be established. A communications session is a logical connection between two systems through which a client program on a local system can communicate with a server program on a remote system.

After a communications session has been established, the client can start the network print server by sending a special record, called a Program Start Request (PSR), to the iSeries server. The communications session you use can utilize different protocols. Refer to the appropriate manual for that communications type for information on using program start requests (PSRs).

After the network print server program has been started using a PSR, a communications transaction has been started. A communications transaction is a logical connection between two programs on a communications session. After this communications transaction starts, data can be exchanged between the client and the network print server.

Prestart Jobs and the Network Print Server

Prestart jobs provide increased performance between a program start request (PSR) initiated by a client and the network print server. Prestart jobs are defined within a subsystem. Prestart jobs become active when that subsystem is started or they can be controlled with the Start Prestart Job (STRPJ) and End Prestart Job (ENDPJ) commands.

The program name contained in the PSR from the client must be defined in the prestart job entry. This is how the PSR attaches itself to a prestart job and therefore achieves better performance.

The network print server has prestart jobs defined in the QBASE and QCMN subsystems. The number of prestart jobs that automatically start to support the network print server is small and thus saves system resources.

Monitoring Prestart Jobs for the Network Print Server

Monitoring prestart jobs for the network print server in the QBASE or QCMN subsystems can be done using the Display Active Prestart Jobs (DSPACTPJ) command. For the network print server, you need to know the subsystem your prestart jobs are in (QBASE or QCMN) and the program the prestart jobs are started for (QNPSERVR).

This command provides the following information:
- Prestart jobs:
    - Current number
    - Average number
    - Peak number
- Prestart jobs in use:
    - Current number
    - Average number
    - Peak number
- Program start requests:
    - Current number waiting
    - Average number waiting
    - Peak number waiting
    - Average wait time
    - Number accepted
    - Number rejected

    Note: The network print server never rejects a PSR. All PSRs are handled, or they are queued and use the next available prestart job.

Changing Prestart Job Entries

The information presented for an active prestart job can be refreshed by pressing the F13 key while on the DSPACTPJ display. Of particular interest is the information about PSRs. This information can indicate to you whether or not you need to modify the available number of prestart jobs. If you have information indicating PSRs are waiting for an available prestart job, you can modify prestart jobs using the Change Prestart Job Entry (CHGPJE) command.

Following is an example of a prestart job entry in the QBASE subsystem for the network print server.

```
SBSD(QSYS/QBASE) +
PGM(QSYS/QNPSERVR) +
USER(QUSER) +
STRJOBS(*YES) +
INLJOBS(1) +
THRESHOLD(1) +
ADLJOBS(3) +
MAXJOBS(*NOMAX) +
JOB(*PGM) +
JOBD(*USRPRF) +
MAXUSE(200) +
WAIT(*YES) +
POOLID(1) +
CLS(QGPL/QCASERVR *CALC *NONE *CALC)
```

If the PSRs were not being acted upon fast enough you could:

- Increase the Initial number of jobs (INLJOBS) parameter value.
- Increase the Additional number of jobs (ADLJOBS) parameter value.

  Then, when the Threshold (THRESHOLD) parameter value is reached, the additional number of jobs are prestarted.

The key is to match the number of prestart jobs to the number of PSRs that are being sent. Keeping this association as close to 1-to-1 ensures peak system performance.

Additional prestart jobs can be started if the number of program start requests to the network print server exceed the number of available prestart jobs.

Exit Points and the Network Print Server

An exit point is a specific point in a system function or program where control may be passed to an installation specified program or programs. These programs are called exit programs because they are providing an exit from the normal processing of the system function or program. The exit programs are usually programs created by users.

Table 16 contains the IBM-registered exit points that can be used with the network print server.

Table 16. Network Print Server Registered Exit Points

Type of Support	Exit Point Name	Exit Point Format
Entry	QIBM_QNPS_ENTRY	ENTR0100
Spooled file	QIBM_QNPS_SPLF	SPLF0100

The functions provided by these exit points are:

- QIBM_QNPS_ENTRY

  This exit point is used by exit programs that want to control which users can access the network print server.
- QIBM_QNPS_SPLF

  This exit point is used by exit programs that process spooled files.

Exit programs must be registered with the OS/400 Registration Facility. Registration allows all users to associate exit programs with the exit points.

OS/400 Registration Facility and the Network Print Server

The registration facility provides storage and retrieval service for both OS/400 and non-OS/400 exit points and exit programs.

The network print server exit points are already registered with the OS/400 Registration Facility. To use these exit points, you have to register your exit programs with the OS/400 Registration Facility.

Exit programs are registered and de-registered using the following APIs:

- Add Exit Program (QUSADDEP) API

  Adds an exit program entry for a specific exit point.

  Note: The network print server exit points do not define exit program data. When you register your exit program you can provide exit program data. However, the data is not used by the network print server.

- Remove Exit Program (QUSRMVEP) API

  Removes an exit program entry for a specific exit point.

For information and examples on how to register exit programs and exit points, see the API Reference topic in the iSeries Information Center.

Verifying Exit Point and Exit Program Registration

The Work With Registration Information (WRKREGINF) command displays information about exit points and exit programs. The command does not add, delete, or change exit points or exit programs within the registration facility. The APIs listed above must be used to initiate change.

Using the OS/400 Network Print Server Exit Points

The diagram below provides an example of several requests from a client to the network print server using the exit points provided by the network print server. Those requests are: access the network print server, call a user-written exit program to verify access, put a spooled output file on an output queue, and call a

user-written exit program that faxes a spooled output file.

```
Client request          NPS program
to access NPS
                Exit point name: QIBM_QNPS_ENTRY        Customer exit program
                      Exit program: CHECKEM             Named: CHECKEM
                               .
                               .
                               .
                         NPS program

Client request                 .
to write a                     .
spooled file             NPS program.                   Spooled output file
in an output                   .                        placed in output
queue                          .                        queue
                               .
                               .
                         NPS program
                               .
                               .
                               .
                       Spooled output file
Client request                 .
to call an
exit program
for a spooled
file

                Exit point name: QIBM_QNPS_SPLF         Customer exit program
                      Exit program: FAXIT               Named: FAXIT
                               .                        Spooled outpu file sent to
                               .                        fax machine
                               .
                         NPS program
                               .

                                                        RBAFT524-0
```

Exit Point QIBM_QNPS_ENTRY

Exit Point QIBM_QNPS_ENTRY is used by exit programs that want to control which users can access the network print server. The called exit program is passed two parameters. These parameters are pointers to structures in user domain storage. These structures, collectively, comprise format ENTR0100.

Required Parameters:

Required Parameter Group:

1	Return code	Output	CHAR(1)
2	Server information	Input	CHAR(32)

Required Parameter Group

Return code

OUTPUT; CHAR(1) The return code parameter is a pointer to a structure in user domain that contains the return code from the exit program. The return code is checked upon return from the call to the exit program.

A return code value of hexadecimal F1 indicates that the network print server allows access and continues processing.

Any other return code value indicates that the network print server rejects access and stops processing. The exit program being called must log any specific errors.

The structure of the return code is:

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code

Server information
INPUT; CHAR(32) The server information structure contains the user profile name, server identifier, format name, and server function identifier.

The structure of the server information is:

Offset			
Dec	Hex	Type	Field
0	0	CHAR(10)	User profile name
10	A	CHAR(10)	Server identifier
20	14	CHAR(8)	Format name
28	1C	BINARY(4)	Function identifier

Exit Point QIBM_QNPS_SPLF

Exit Point QIBM_QNPS_SPLF is used by exit programs that process spooled files. The called exit program is passed four parameters. These parameters are pointers to structures in user domain storage. These structures, collectively, comprise format SPLF0100.

Required Parameters:

Required Parameter Group:

1	Return code	Output	CHAR(1)
2	Server information	Input	CHAR(32)
3	Spooled file ID	Input	CHAR(40)
4	Spooled file exit program data	Input	CHAR(*)

Required Parameter Group

Return code
OUTPUT; CHAR(1) The return code is a pointer to a structure in user domain that contains the return code from the exit program. The return code is checked upon return from the call to the exit program.

Any non-zero code indicates an error with the exit program. The exit program being called must log any specific errors. Processing by the network print server continues.

Offset			
Dec	Hex	Type	Field
0	0	CHAR(1)	Return code

Server information

    INPUT; CHAR(32) The server information structure contains the user profile name, server identifier, format name, and server function identifier.

The structure of server information is:

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	CHAR(10)	User profile name
10	A	CHAR(10)	Server identifier
20	14	CHAR(8)	Format name
28	1C	BINARY(4)	Function identifier

Spooled file ID

    INPUT; CHAR(40) The spooled file ID structure contains the job name, user name, job number, spooled file name, and spooled file number. This information uniquely identifies a spooled file on the OS/400.

The structure of a spooled file ID is:

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	CHAR(10)	Job name
10	A	CHAR(10)	User name
20	14	CHAR(6)	Job number
26	1A	CHAR(10)	Spooled file name
36	24	BINARY(4)	Spooled file number

Spooled file exit program data

    INPUT; CHAR(*) The spooled file exit program data information structure contains a four byte length and the spooled file exit program data. The contents of the spooled file exit program data is not known to the network print server. The receiving exit program knows the structure of the spooled file exit program data. For example, if the exit program was going to fax the spooled file specified by the spooled file ID. The spooled file exit program data could consist of a name, phone number, building number, and office location.

The structure of the spooled file exit program data is:

Offset			
Dec	**Hex**	**Type**	**Field**
0	0	BINARY(4)	Length of the spooled file exit program data
4	4	CHAR(*)	Spooled file exit program data

Parameter Field Descriptions

Format name. Name of the format being used.

For the network print server, two formats are supported:
- For spooled file support the value is SPLF0100.
- For entry support the value is ENTR0100.

Function identifier. Identifies, within a server, the function of the exit point. For the network print server, two function identifiers are supported:
- For spooled file support, the value is hexadecimal 010D.
- For entry support, the value is hexadecimal 0802.

Job name. The name of the job that created the spooled file.

Job number. The number of the job that created the spooled file.

Server identifier. Identifies the OS/400 server being called. The value for the network print server is QNPSERVR.

Spooled file exit program data. Spooled file exit program data consists of additional information used by the exit program that has registered for exit point QIBM_QNPS_SPLF. The client application provides the spooled file exit program data.

Spooled file name. The name of the spooled file being requested.

Spooled file number. The number of the spooled file being requested.

User profile name. The user, from iSeries Access for Windows, that is making the call to the network print server.

Part 3. Advanced function printing (AFP)

Chapter 6. What Is Advanced Function Printing (AFP)?

Note: Regarding AFP and Print Services Facility/400 (PSF/400)

To use advanced function printing (AFP) support on the iSeries server, PSF/400 must be installed. However, if the ASCII data stream is converted through Host Print Transform function, you can use the AFP support without having to install the PSF/400.

See "Using Print Services Facility™ for OS/400® (PSF/400)" on page x for information on when PSF/400 is required. If you have additional questions about PSF/400, contact your IBM representative.

Advanced function printing (AFP) is the ability of programs to use the all-points-addressable concept to print text and images on a printer. AFP supports Advanced function printing data stream (AFPDS), Intelligent printer data stream (IPDS) and SNA character string (SCS).

Frequently used terms

Before using this part of the manual, familiarize yourself with these terms, which are used throughout the manual.

Term   Definition of Term

AFCCU™
: Advanced Function Common Control Unit

AFP   Advanced function printing (AFP)

The ability of programs to use the all-points-addressable concept to print text and images on a printer.

PCL   Printer Control Language

PPDS   Personal Printer Data Stream

APA   all-points-addressable

In AFP support, pertaining to the capability to address, refer to, and position text, overlays, and images at any defined point on the printable area of the paper.

AFPDS
: Advanced function printing data stream (AFPDS)

In AFP support, the printer data stream used for printing advanced function printing data. The AFPDS includes composed text, page segments, electronic overlays, form definitions, page definitions, and fonts that are downloaded from the iSeries server to the printer.

APPC/PC
: Advanced Program-to-Program Communications/Personal Computer Program (APPC/PC)

A licensed program, installed on a personal computer system, that performs APPC functions for a transaction program running on the personal computer system.

RPM   Remote PrintManager™ (RPM).

In AFP support, a personal computer product that allows selected font data, overlays, and page segments that are present in advanced function printing data streams to be available to an attached IBM page printer.

PSF for OS/2

Print Services Facility for OS/2 (PSF for OS/2) is a PS/2*-based print server that provides distributed printing in a network environment. Systems can be a host system (S/390*) or several iSeries servers.

PSF for AIX

Print Services Facility for AIX (PSF for AIX) is an RS/6000-based print server that provides distributed printing in a network environment. Systems can be a host system (S/390*) or several iSeries servers.

PSF/400

Print Services Facility/400

DPF   Distributed print function

DPF is a part of PSF for OS/2. It permits host and iSeries servers to simultaneously send IPDS spooled files to PSF for OS/2. The files are first spooled onto the PS/2's or RS/6000's hard drive before being printed.

PSF Direct

Print Services Facility Direct (PSF Direct) is the function in PSF for OS/2 and PSF for AIX that replaces the pass-through function of Remote PrintManager (RPM) 2.0. It permits starting multiple printer writers from iSeries servers or other hosts to printers attached to PSF for OS/2 or PSF for AIX Each printer writer has direct control of the printer to which it is attached. OS/400 spooled files are not stored on the PC's hard drive.

Advanced Function Printing Data Stream (AFPDS)

AFPDS files can be generated on the iSeries server as well as on a System/390*. The ability of both systems to generate the AFPDS data stream means that they can send AFPDS files to each other for printing on AFP-configured printers.

On the iSeries server, AFPDS is generated by selecting *AFPDS as the value on the DEVTYPE parameter of the printer file being used.

What this part of the manual will do for you

Depending on your AFP environment, tasks to be completed could be: install hardware, configure printers, configure personal computer systems, and create configuration descriptions on the iSeries server using the following manuals:
- The iSeries server library manuals
- IBM personal computer manuals
- IBM token-ring manuals
- IBM TCP/IP manuals
- IBM printer manuals
- PSF for AIX manuals for PSF for AIX configuration
- PSF for OS/2 manuals for PSF for OS/2 configuration
- The *IBM Remote PrintManager User's Guide and Installation Guide*
- The *i-data 7913-03 IPDS Printer LAN Attachment - Installation Guide*
- The *i-data 7913-04 IPDS Printer LAN Attachment - Installation Guide*

This part of the guide organizes all of these tasks and directs you, when necessary, to the correct manual required to complete a task.

Chapter 7. Resources needed to perform Advanced Function Printing (AFP)

This chapter discusses the resources used to perform advanced function printing.

Some of the resources are shipped with the system and some of them can come from another system. You can download resources from another system, or they can come inline with the data that is to be printed. Form definitions, fonts, page segments, page overlays, and page definitions are examples of resources that can come from another system. Additionally, you could have created page overlays and page segments yourself using the Advanced Function Printing Utilities/400 (AFP Utilities) licensed program (5769-AF1). This licensed program allows you to create resources for advanced function printing on the iSeries server. For more information on AFP Utilities, go to "Advanced Function Printing Utilities/400" on page 313.

If you are certain you have all the resources needed to use AFP, you can skip this chapter and go to Chapter 9, "Printing AFPDS data" on page 225. If you do not have the resources, or are unsure, read this chapter.

Resource objects contain data and control information that can be used in a printing job and that can be shared by different pages and different print data sets. Examples of resources are fonts, which define the characters used to print text, and page segments, which can include images and text.

Resources can be stored and accessed from within the operating system being used. By referring to the name of the stored resource, many data streams can share the same resources.

When you received AFPDS resource data from a system other than an iSeries server, you transformed that data to a format that can be used on the iSeries server. You did that by using the following commands:

- CRTFORMDF
- CRTFNTRSC
- CRTPAGSEG
- CRTOVL
- CRTPAGDFN

When the commands are run, it is important that the transformed data be placed in a **library** that is defined to your **library list** when you initially sign on the system. This ensures that System/390 resources are available when the spooled file is created and when it is printed.

A **library** is an object on disk that serves as a directory to other objects. A **library list** is a list that indicates which libraries are to be searched and the order in which they are to be searched.

When specifying an overlay, page definition, or page segment it is important to have the library they are placed in on your library list. And, if you have more than one overlay, page definition, or page segment with the same name, make sure the

one you want to use is in the first library searched. Otherwise, you could specify the correct name, but not print with the correct resources.

Working with AFP resources and libraries

When the spooled file is created, the system saves the libraries in your library list. Within that list are the libraries that contain the resources used to create the spooled file.

When the system searches the saved libraries for the resources needed to print the spooled file, it searches in the following order:

1. System libraries

   These libraries are identified in system value QSYSLIBL. System libraries are available to all users; they are identified in your library list with a type of SYS.

2. Current library

   The current library is identified in your user profile as the value for the current library (CURLIB) parameter. In your library list, the current library has a type of CUR.

3. User libraries

   User libraries are identified in system value QUSRLIBL or a job description. In your library list, these libraries are identified with a type of USR.

If you have a resource that you want many users to have access to (fonts, for example), you may want to put that resource in a library that everyone has access to, such as a library identified in the system value QSYSLIBL.

On the other hand, if you have a resource that you want to restrict access to (signatures, for example), you may want to put that resource in a library that is accessible only to the users that need that resource. For example, if you own the library that the resources are stored in, you can grant access to that object (the library) with the Grant Object Authority (GRTOBJAUT) command.

You may override the above system library search list and specify a library search list for a particular printer with the PSF configuration object. The PSF configuration object allows you to specify additional parameters for a particular device that are not supported on the Create Device Description (CRTDEVPRT) command. This includes the ability to specify a user library list (USRRSCLIBL parameter) and device library list (DEVRSCLIBL parameter). If you create a PSF configuration object, the default for the user and device library lists will be the same as the library search described above. For more information about specifying AFP resource library lists with a PSF configuration object, see "User and device resource library lists" on page 246.

Fonts and font libraries

When a printer writer is started for an AFP-configured printer, the system searches for the specified font in the user's library list and then the IBM-supplied font libraries. The IBM-supplied font libraries are:

- QFNTCPL

  This library contains the OS/400 compatibility fonts. These are the fonts shipped with the iSeries server.

- SBCS font libraries numbered QFNT01 through QFNT19 and DBCS font libraries numbered QFNT61 through QFNT69.

These are the font libraries that support various IBM licensed programs on the iSeries server.

Any of those libraries that are found are put in the library list of the job printing the spooled file. These font resource libraries are then available for printing spooled files even though they were not in the library list of the job (user) that originally created the spooled file. Also, if you have acquired additional font resources, you could store them in these IBM-supplied font libraries. Storing them in the IBM-supplied libraries provides widespread access based on the way the system searches for a font.

Putting the font resource in a separate library that is not one of the QFNTxx libraries listed above requires that users have that library specified in their library list when the spooled file is created.

For information about specifying a different font library list on a printer, see "User and device resource library lists" on page 246.

Font resource objects

A font is a collection of characters of a certain size, typeface, and type style. Each character in a font is identified by a 1-byte or a 2-byte code. The internal structure of fonts depends on whether the font is for a phonetic writing system, like English, or for a nonphonetic writing system, like Kanji.

At least two resources are needed to make up a font: a font character set and a code page. A third resource, a coded font, can define a font by naming a font character set and a code page.

Font character set
This resource contains the raster patterns for each character in the font and associates an 8-byte character identifier with each pattern. This resource also contains descriptive information for the entire character set.

Code page
This resource associates code points with character identifiers, each representing a character raster pattern. A code point is an 8-bit binary number representing one of 256 potential characters.

Coded font
This resource associates one or more code pages with the appropriate font character sets.

Fonts from System/390

Fonts may be sent to the iSeries server from the System/390. Check your licensing agreements to see if they preclude the transfer of printer resources between systems.

Font resources can be downloaded from the System/390. These font resources can not be altered by the iSeries server operator. If changes need to be made to any font resources the System/390 site must be notified.

When these font resources are received from the System/390, you place them in a data file. In order to convert these font resources to a format that can be used by OS/400, you use the **CRTFNTRSC** (Create Font Resource) command.

3800 Printer Font Restrictions on the iSeries server:

You can receive data from a System/390 that was generated to print on a 3800 printer using 3800 printer font character sets.

However, the iSeries server does <u>not</u> support the 3800 printer and its font character sets.

Do not receive 3800 printer font character sets on the iSeries server.

If you have AFPDS data that calls for a 3800 font and you have the 3800 printer font character sets on the system, the system will try to use them and you will get an error when the data prints.

Instead, download the 3820 printer font character sets from the System/390. When the data generated for the 3800 printer starts to print, OS/400 will substitute the 3820 printer font character sets.

The naming convention for 3800 printer font character sets is: C1xxxxxx. The naming convention for 3820 printer font character sets is: C0xxxxxx.

Fonts provided by the OS/400 program

The compatibility set is provided to allow OS/400 applications to print on the 3820, 3825, 3827, 3828, 3829, 3831, 3835, and 3900 Model 1 Printers. The numeric font identifiers specified in the application are translated into one of the fonts in the compatibility set.

The following font families comprise the IBM-supplied compatibility font set that is provided with the OS/400 program for advanced function printing:

- APL
- Boldface
- Courier
- Document
- Essay
- Format
- Gothic
- Orator
- Prestige
- Proprinter Emulation
- Roman
- Script
- Serif
- Symbol Set
- Text

In addition to these 240-pel raster fonts, the compatability set includes a number of code pages that are downloadable to printers. For use on printers that support outline fonts, the set also includes a Letter Gothic outline font that contains the euro symbol.

When your operating system was loaded, you had to choose to install or not install IBM-supplied fonts. If you chose not to install the fonts at that time, and now wish to install them, use the Software Install PDFl to install the fonts.

Advanced Function Printing Fonts/400 (Program 5769-FNT)

The licensed program Advanced Function Printing Fonts/400 contains all the fonts listed below. They can also be purchased individually.

- Sonoran Serif** [1]
- Sonoran Serif Headliner
- Sonoran Sans Serif** [2]
- Sonoran Sans Serif Headliner
- Sonoran Sans Serif Condensed
- Sonoran Sans Serif Expanded
- Monotype Garamond**
- Century Schoolbook**
- Pi and Specials
- ITC Souvenir**
- ITC Avant Garde Gothic**
- Mathematics and Science
- Optical Character Recognition (OCR-A and OCR-B)
- DATA1
- APL2®

If you purchase any of the fonts and want to install them, you will have to use the Software Install PDF to install the fonts.

For information on font substitution go to Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377.

AFP font collection

Version 2.1.0 of the AFP Font Collection is similar to the first version (1.1.0). It also adds support for the Euro currency symbol to the Latin1, Thai language, and Lao language fonts, as well as some new code pages. Version 2.1.0 divides the font collection into two categories:

- IBM AFP Font Collection for MVS, OS/390®, VM, and VSE
- IBM AFP Font Collection for AIX®, OS/2, and OS/400

IBM AFP font collection for MVS, OS/390, VM, and VSE:
This font collection provides font resources for the MVS, OS/390, VM, and VSE operating environments. It includes font data formats for 240-pel bitmaps, 300-pel bitmaps, and AFP outline fonts. No additional font utilities are offered with this collection.

IBM AFP font collection for AIX, OS/2, and OS/400:
This font collection provides font resources for the AIX, OS/2, and OS/400 operating environments. In addition to the font resources, additional font utilities are available that allow you to edit outline fonts, create AFP font resources from Type 1 outline fonts, create and modify code pages, and create and modify coded fonts.

---

1. Sonoran Serif is a functional equivalent of Monotype Times New Roman.

2. Sonoran Sans Serif is a functional equivalent of Monotype Arial.

Chinese, Japanese, and Korean languages have DBCS CID-keyed outline fonts. Fonts for these three languages are also available for Type Transformer and related OS/2 batch jobs to create 240-pel and AFP Outline Font resources.

AFP expanded core fonts

The Expanded Core Fonts are the SBCS type families provided in the AFP Font Collection. They include:

- Boldface Latin-1 (with Euro)
- BookMaster® Latin-1 and Specials (with Euro)
- Courier (with Euro)
- Courier APL2
- Gothic Katanka
- Gothic Text Latin-1 (with Euro)
- Helvetica (with Euro)
- IBM Logo (for BookMaster)
- Letter Gothic Latin-1 (with Euro)
- OCR-A
- OCR-B
- Prestige Latin-1 (with Euro)
- Times New Roman (with Euro)

Language support includes the Latin-1–5 language groups, along with Arabic, Cryllic, Greek, Hebrew, Lao and Thai for the Courier, Helvetica, and Times New Roman type families.

DBCS type families and languages provided in the AFP Font Collection for Outline Fonts and Programs include:

- Japanese
  - Heisei Kaku Gothic
  - Heisei Maru Gothic
  - Heisei Mincho
- Korean with unified Hanguel characters
  - Gothic
  - Myengjo
- Chinese-Simplified
  - Fang Song (GB)
  - Hei (GB)
  - Kai (GB)
  - Song (GBK)
- Chinese-Traditional
  - Kai
  - Sung

The Version 2.1.0 AFP Font Collection does not contain any of the following:

- Font licensed products, including the Sonoran fonts.
- Font RPQs, including Postal Bar Codes and Bar Code/OCR
- 4028 Font Metrics
- Compatibility fonts in 5648-B33; however, they are included in 5648-B45.

Supported languages

The AFP Font Collection products provide language support as follows:

Note: Language groups identified in items 1 through 9 are defined in the International Organization for Standardization (ISO) standard 8859. Not every font provides characters for every language cited.

1. The Latin/Arabic language group contains characters for Latin and Arabic scripts.
2. The Latin/Cyrillic language group contains characters for Bulgarian, Byelorussian, Macedonian, Russian, Serbian, Ukrainian, and English.
3. The Latin/Greek language group contains characters for Latin and Greek scripts.
4. The Latin/Hebrew language group contains characters for Latin and Hebrew scripts.
5. The Latin-1 language group contains characters for the following languages: Danish, Dutch, English, Faeroese, Finnish, French, German, Icelandic, Irish, Italian, Norwegian, Portuguese, Spanish, and Swedish.
6. The Latin-2 language group contains characters for the following languages: Albanian, Czech, English, German, Hungarian, Polish, Romanian, Serbocroatian, Slovak, and Slovenian.
7. The Latin-3 language group contains characters for the following languages: Afrikaans, Catalan, Dutch, English, Esperanto, French, German, Italian, Maltese, Spanish, and Turkish.
8. The Latin-4 language group contains characters for the following languages: Danish, English, Estonian, Finnish, French, German, Greenlandic, Lappish, Latvian, Lithuanian, and Norwegian.
9. The Latin-5 language group contains characters for the following languages: Danish, Dutch, English, Finnish, French, Irish, Italian, Norwegian, Portuguesse, Spanish, Swedish, and Turkish.
10. The Latin/Thai language group contains characters for Latin and Thai scripts.
11. The Latin/Lao language group contains characters for Latin and Lao scripts.
12. Katakana/Gothic Katakana contains phonetic syllabic characters used for writing non-Japanese words and for emphasis in Japanese.
13. The DBCS Fonts contain characters for Simplified Chinese, Traditional Chinese, Japanese, and Korean.

IBM compatibility fonts:

These fonts consist of uniformly spaced and mixed-pitch type families such as Courier, Document, Essay, Letter Gothic, Orator, and Prestige. The Proprinter Emulation fonts are also included, which are used with the supported printers emulation mode.

The IBM Compatibility Fonts are provided in 240-pel bounded and 300-pel raster formats.

IBM 4028 font metrics:

These font metrics allow AFP applications to format documents to be printed on the 300 dots-per-inch IBM LaserPrinter 4028 using printer resident fonts.

Advanced Function Printing DBCS Fonts/400 (Program 5769-FN1)

The licensed program Advanced Function Printing DBCS Fonts/400 contains all the fonts listed below. Each of the five features can be ordered separately.

- AS/400 Font DBCS - Japanese
- AS/400 Font DBCS - Korean
- AS/400 Font DBCS - Traditional Chinese
- AS/400 Font DBCS - Simplified Chinese
- AS/400 Font DBCS - Thai

Page segments

Page segments are objects containing composed text and image, prepared before formatting and included during printing.

For example: you may want to repeat constant data at different positions on a page or overlay, and you may also want to repeat that data on different pages or overlays. You can do this by using a page segment. A company logo is an example of this type of data.

Using the Advanced Function Printing Utilities/400 licensed program or the AFP printer drivers provided with Microsoft® Windows® or OS/2 2.0, you can create page segments on the iSeries server. Page segments can also be downloaded from System/390. The page segments must be stored in an accessible library.

When page segment data is received from the System/390, you place it in a data file. In order to convert the page data to a format that can be used by OS/400, use the **CRTPAGSEG** (Create Page Segment) command.

Overlays

Overlays are a collection of predefined data (such as lines, shading, text, boxes, or logos) that can be merged with variable data on a sheet while printing is being performed.

To use overlays, specify them in the front and back overlay parameters of the printer file being used with your application: temporarily using an Override with Printer File (OVRPRTF) command, permanently using the Change Printer File (CHGPRTF) command, or before the spooled file prints using the Change Spooled File Attributes (CHGSPLFA) command.

You can also use the DDS OVERLAY keyword to include overlays with your printed output.

The licensed program Advanced Function Printing Utilities/400 (5738-AF1) or the AFP printer drivers provided with Microsoft Windows or OS/2 2.0 provide the capability to create overlays on the iSeries server. Page overlays can also be downloaded from the System/390.

The overlays must be stored in an accessible data file.

When overlay data is received from the S/390, you place it in a data file. In order to convert the page overlay data to a format that can be used by OS/400, use the **CRTOVL** (Create Overlay) command.

Form definitions

A form definition is an AFP resource object that defines the characteristics of the form including:

- Overlays
- Position of page data on the form
- Rotation
- Duplexing
- Input drawer
- Formfeed type
- Print quality
- Number of copies of each page and the modifications that apply to each set of copies.

Whenever you are printing to a printer configured as AFP(*YES), a form definition is required. If a form definition is not specified on the printer file, an inline form definition is built from the printer file parameters.

The following list details the origin of AFP-related form definitions:

- Provided with the iSeries server
- Created with AFP PrintSuite for OS/400.
- Downloaded from S/390
- Inline from S/390

Form definitions created with AFP PrintSuite for OS/400

Form definitions can be created on the OS/400 with the Page Printer Formatting Aid (PPFA/400) feature of the AFP PrintSuite for OS/400. The Convert Page Printer Formatting Aid Source (CVTPPFASRC) command that is a part of PPFA/400 allows you to create a source file containing form definition structured fields. You would then use the Create Form Definition (CRTFORMDF) command to create a form definition from the source file generated by the CVTPPFASRC command. See the *IBM Page Printer Formatting Aid: User's Guide (S544-5284)* for more information about creating form definitions on OS/400.

Form definitions provided with the iSeries server

The following form definitions are provided with the iSeries server in library QSYS.

Name	Across (Inches)	Down (Inches)	Presentation	Direction	Bin	Duplex
F1A10110	1/6	1/6	Portrait	Across	1	No
F1A10111	1/6	1/6	Portrait	Across	1	Yes
F1A10112	1/6	1/6	Portrait	Across	1	Tumble
F1A10120	1/6	1/6	Portrait	Across	2	No
F1A10121	1/6	1/6	Portrait	Across	2	Yes
F1A10122	1/6	1/6	Portrait	Across	2	Tumble
F1C10110	1/6	1/6	Landscape	Down	1	No
F10101PA	0	1/2	Portrait	Across	1	No
F10101PD	0	1/2	Portrait	Down	1	No

Name	Across (Inches)	Down (Inches)	Presentation	Direction	Bin	Duplex
F10101LA	0	1/2	Landscape	Across	1	No
F10101LD	0	1/2	Landscape	Down	1	No
F1OGL	0	0	Portrait	Across	1	No

Note: The IBM-supplied form definitions do not set the print quality for the 4224, 4230, 4234, and 4247 printers. You must set the print quality on the printer operator panel when using an IBM-supplied form definition.

Form definitions downloaded from System/390

Form definitions can also be downloaded from the System/390. These form definitions cannot be altered by the iSeries server operator. If changes need to be made to any form definition, the System/390 site must be notified.

When these form definitions are received from System/390, you place them in a data file. In order to convert this form definition data to a format that can be used by OS/400, use the **CRTFORMDF** (Create Form Definition) command.

Form definitions that are inline from System/390

Form definitions can also be part of the AFPDS file that is sent to the iSeries server. That is, the form definition and the data all come in one large data stream. Here again, if any change is needed to the form definition, the System/390 site must be notified.

FORMDF parameter on printer file

To use or identify a fully-qualified form definition, specify the FORMDF parameter with the CRTPRTF, CHGPRTF or OVRPRTF CLcommands.

You can specify a form definition with the following device types:
- *AFPDS
- *AFPDSLINE
- *LINE
- *IPDS
- *SCS
- *USERASCII

Whenever you specify any output directed to an AFP printer but do not specify a form definition, an inline form definition is built from the print parameters and passed to PSF/400.

Depending upon the specifications given, some printer file parameters may be ignored when output is printed through PSF/400. For example, if you specify a form definition on the CRTPRTF, CHGPRTF or OVRPRTF CL command, an inline form definition will not be built from the printer file parameters. In this example, if you send the data to an AFP printer PSF/400 ignores the following printer file parameters:
- BACKMGN
- CORNERSTPL
- DRAWER
- DUPLEX

- EDGESTITCH
- FORMFEED
- FRONTMGN
- MULTIUP (N_UP)
- PAGRTT
- PRTQLTY
- REDUCE (N_UP)
- SADLSTITCH

However, if you send the data to a non-AFP printer with a devtype of *IPDS or *SCS in this example, the form definition parameter is ignored and the printer file parameters are used.

When specifying a form definition on the printer file, the values you specify on the DRAWER and DUPLEX parameters will override the drawer and duplex values specified in the form definition. If you want to use the drawer and duplex values specified in the form definition, you must specify DRAWER(*FORMDF) and DUPLEX(*FORMDF) on the printer file.

To indicate an output bin for a form definition, specify the OUTBIN parameter on the print file.

The spooled file attributes for the form definition are changeable.

Page definitions

Page definitions are resources that format and compose line data into pages. A page definition contains printing controls that specify:
- Where data from each input record is to be printed
- Page size (height and width)
- Data fields that can be suppressed
- Print positions for line-data records containing carriage-control characters
- Inline printing direction
- Number of lines per inch
- List of page segments that may be used
- List of overlays that may be used
- Record definitions
- Constant data to be printed
- List of fonts that may be used

Whenever you are printing to a printer configured as AFP(*YES), and you are printing line data or mixed data, a page definition is required. If a page definition is not specified on the printer file, an inline page definition is built from the printer file parameters.

The page definitions shown in Table 17 are provided with the iSeries server in library QSYS.

Table 17. IBM-Supplied Page Definitions

Name	Size of Form	Orientation on Page	Description
P1A06462	8.5 x 11.0 inches	Portrait	Letter size. 6 lines per inch (LPI). 64 lines per page
P1A08584 [1]	9.5 x 11.0 inches	Portrait	Continuous forms. 8 lines per inch (LPI). 85 lines per page.
P1A08682	8.5 x 11.0 inches	Portrait	Letter size. 8 lines per inch (LPI). 86 lines per page.
P1B08262	8.5 x 14.0 inches	Portrait	Legal size. 6 lines per inch (LPI). 82 lines per page.
P1B04963	8.5 x 14.0 inches	Portrait	Legal size. 6 lines per inch (LPI). 82 lines per page.
P1B11082	8.5 x 14.0 inches	Portrait	Legal size. 8 lines per inch (LPI). 110 lines per page.
P1B06683	14.0 x 8.5 inches	Landscape	Legal size. 8 lines per inch (LPI). 66 lines per page.
P1C09182	8.27 x 11.69 inches	Portrait	Legal size. 8 lines per inch (LPI). 91 lines per page.
P1D08462	14.33 x 10.12 inches	Landscape	B4 size forms. 6 lines per inch (LPI). 84 lines per page.
P1D08083	14.33 x 10.12 inches	Landscape	B4 Size forms. 8 lines per inch (LPI). 80 lines per page.
P1D11382	10.12 x 14.33 inches	Portrait	B4 Size forms. 8 lines per inch (LPI). 113 lines per page.
P1J04964 [1]	12.0 x 8.5 inches	Landscape	Continuous forms. 6 lines per inch (LPI). 49 lines per page.
P1J06484 [1]	12.0 x 8.5 inches	Landscape	Continuous forms. 8 lines per inch (LPI). 64 lines per page.
P1L06464 [1]	14.88 x 11.0 inches	Landscape	Continuous forms. 6 lines per inch (LPI). 64 lines per page.
P1L08584 [1]	14.88 x 11.0 inches	Landscape	Continuous forms. 8 lines per inch (LPI). 85 lines per page.
P1V04863	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms. 6 lines per inch (LPI). 48 lines per page.
P1V06483	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms. 8 lines per inch (LPI). 64 lines per page.
P1V06683	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms. 8 lines per inch (LPI). 66 lines per page.

Table 17. IBM-Supplied Page Definitions (continued)

Name	Size of Form	Orientation on Page	Description
P1W240F3	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms using MULTIUP. 15 lines per inch. 240 lines printed per page. 4 logical pages printed on one physical page.
P1W120C2	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms using MULTIUP. 12 lines per inch. 120 lines printed per page. 2 logical pages printed on one physical page.
P1W12883	8.27 x 11.00 inches	Portrait	Combined letter/A4 size forms using MULTIUP. 8 lines per inch. 128 lines printed per page. 2 logical pages printed on one physical page.
Note:			
1 Can only be used with the 4224, 4230, 4234, and 4247 printers.			

When page definition data is received from the S/390, you place it in a data file. To convert the page definition data to a format that can be used by OS/400, use the **CRTPAGDFN** (Create Page Definition) command.

Page definitions can be created on OS/400 with the Page Printer Formatting Aid (PPFA/400) feature of AFP PrintSuite for OS/400. The CVTPPFASRC command that is a part of PPFA/400 allows you to create a source file containing page definition structured fields. You would then use the CRTPAGDFN command to create a page definition from the source file generated by the CVTPPFASRC command. See the *IBM Page Printer Formatting Aid: User's Guide (S544-5284)* for more information about creating page definitions on OS/400.

PAGDFN parameter on printer file

To use or identify a fully-qualified page definition, specify the PAGDFN parameter with the CRTPRTF, CHGPRTF, or OVRPRTF CL commands.

You can specify page definitions with *LINE or *AFPDSLINE data. After PSF/400 completes formatting, it converts the line data and page definition to IPDS.

When you specify a page definition on the printer file, some printer file parameters may be ignored when the spooled file is printed through PSF/400. For example, if you specify a page definition on the CRTPRTF, CHGPRTF, or OVRPRTF CL command, and also specify line data or mixed data, an inline page definition will not be built from the printer file parameters. In this case, if you send the data to an AFP printer, PSF/400 ignores the following print parameters:

- CDEFNT
- CHRID
- CPI
- FNTCHRSET

- FOLD
- FONT
- LPI
- MULTIUP
- PAGESIZE
- PAGRTT
- REDUCE

However, in this example, if you send the data to a non-AFP printer with a devtype of ***LINE**, the page definition parameter is ignored and the print parameters are used. The line data is converted to SCS or IPDS.

To change the PAGDFN spooled file attribute, use the CHGSPLFA command.

What to do next:

Once you are sure you have all the above resources in place, go to Chapter 9, "Printing AFPDS data" on page 225

Chapter 8. Working with line data

Advanced Function Printing (AFP) applications can generate Advanced Function Printer Data Stream (AFPDS), Line, or Mixed (AFPDS and line) spool files. This chapter describes line and mixed data stream support.

Definitions of line data and mixed data

- Line data

  Line data is record oriented readable text. The generating application can partially format line data by adding carriage control characters to the first column of each record, by adding blank lines to adjust vertical positioning, or by using skipping and spacing control in their applications (such as on a RPG output specification). Line data is supported as a device type in the print file commands (CRTPRTF, CHGPRTF, and OVRPRTF) through the DEVTYPE(*LINE) specification.

- Mixed data

  Mixed data is line data intermixed with AFPDS data. Only certain AFPDS structured fields are allowed to be intermixed with line data. Refer to *Advanced Function Printing: Programming Guide and Line Data Reference*, (S544-3884), for more information about mixing line data and AFPDS structured fields.

For line data and mixed data, the following terms are used interchangeably:

- *LINE and line data
- *AFPDSLINE and mixed data

The following sections provides information about line data application considerations, device type considerations, OS/400 printer file parameters, carriage control characters (ANSI and Machine), Table Reference Characters, IGC parameters, Medium-Map-Name (INVMMAP) DDS keyword, restrictions when using PAGDFN and FORMDF.

DEVTYPE values

To place line data or mixed data onto the printer spool, specify either the *LINE or *AFPDSLINE values with the DEVTYPE parameter for the CRTPRTF, CHGPRTF, and OVRPRTF CL commands.

*LINE Line data is placed onto the spool. For *LINE:, specify any of the following:

- CTLCHAR(*FCFC)
- CTLCHAR(*MACHINE)
- CTLCHAR(*NONE)

To place line data and skipping or spacing controls directly onto the printer spool without converting it to another data stream, specify *LINE. The line data is not in a printer-ready format and, like AFPDS, will be converted to the appropriate printer format at print time.

*AFPDSLINE:

Mixed data (line and AFPDS data) is placed onto the printer spool.

For *AFPDSLINE, specify any of the following:

- CTLCHAR(*FCFC)
- CTLCHAR(*MACHINE)
- CTLCHAR(*NONE)

You can specify page definitions to format traditional application line data without making any application programming changes. If, however, you want to use any one of the following AFPDS structured fields (which can be intermixed with line data), you must specify, in hex, a X'5A' record in the output buffer. Refer to *Advanced Function Printing: Programming Guide and Line Data Reference* , (S544-3884), for more information about mixed documents and AFPDS. For example: by changing your application, you can specify any one of the following AFPDS structured fields, and they can be intermixed with line data:

- Invoke Data Map
- Invoke Medium Map
- Invoke Page Segment
- Include Page Overlay
- Presentation Text

Refer to the *Advanced Function Presentation*™ *Programming Guide and Line Data Reference* (S544-3884) for more information about mixed data streams.

CTLCHAR values

When machine code control characters exist in the data (rather than ANSI control characters), specify the CTLCHAR parameter with a value of *MACHINE on the CRTPRTF, CHGPRTF, or OVRPRTF CL commands.

See Table 19 on page 219 for information about machine code control characters.

TBLREFCHR parameter

To indicate whether a table reference character (TRC) exists in the data, specify the TBLREFCHR parameter with the CRTPRTF, CHGPRTF, or OVRPRTF CL commands.

If forms control characters are used with the data, the TRC follows the forms control character, but precedes the data bytes. If forms control characters are not used, the TRC is the first byte of the data record. As with forms control characters, if table reference characters are used, every data record must contain a TRC byte.

The TBLREFCHR parameter provides support for applications migrating from S/390 to OS/400.

Note: The TBLREFCHR parameter is ignored if specified for *USERASCII, *SCS, *IPDS, and *AFPDS device types.

See "Table reference characters (TRC)" on page 221 for more information.

AFPCHARS parameter

To identify up to four 4-byte names of coded fonts, specify the AFPCHARS parameter with the CRTPRTF, CHGPRTF, or OVRPRTF CL commands. The 4-byte names are concatenated to 'X0' to identify up to four coded fonts that are needed when TBLREFCHR is used within the data.

The AFPCHARS parameter is only valid for *LINE and *AFPDSLINE device types.

CVTLINDTA parameter

Use the CVTLINDTA parameter to specify whether line data and a page definition should be converted to AFPDS before the data is spooled. You can specify this parameter in the CRTPRTF, CHGPRTF, and OVRPRTF commands.

Application considerations for line data

Line data and mixed data on OS/400 are used by S/370™ AFP users who are migrating data to iSeries server users who are generating AFPDS, IPDS, or SCS data streams.

If you are a S/370 user, you should be familiar with the concepts of AFP and page definitions. When using line data, page definitions, and form definitions, your applications are generating line data with either an ANSI or machine code control character in column 1 of the spooled output. To migrate data to AS/400 prior to OS/400 Version 3, Release 2, and Version 3, Release 7, S/370 users had to use PrintManager/400 API calls to rewrite their applications. With line data support, applications do not need to be rewritten. To migrate data to OS/400, specify CTLCHAR(*FCFC) for ANSI code control, or CTLCHAR(*MACHINE) for machine code control. You should also specify DEVTYPE(*LINE) or DEVTYPE(*AFPDSLINE)

If you are an OS/400 user, and want to use line data, page definitions, and form definitions, you must determine whether your application generates an ANSI control character exists in column 1 of your spooled output.

If your application does generate an ANSI control character in column 1 of your spooled output to control skipping and spacing, specify CTLCHAR(*FCFC) on the printer file. Also, to convert to line data, specify the following in the OS/400 printer file:
- DEVTYPE(*LINE)
- PAGDFN (a page definition)
- FORMDF (a form definition) optional format

Note: Specifying a form definition in this format is optional; you could specify that an inline form definition be built from the print file parameters by indicating FORMDF(*NONE) on the printer file.

You can change your application to place an ANSI control character in column 1 of your spooled output to control skipping and spacing, by using a language or application construct (such as a SKIP or **SPACE** option on a COBOL WRITE statement), or by making an RPG output specification.

When you specify a device type of *AFPDS, *IPDS, or *SCS, control information is used to generate the appropriate skipping or spacing commands in the specified

data stream. The control information for *AFPDS and *LINE that is passed by the compilers and application is converted to a machine code control character. Thus, applications that do not use ANSI control characters can generate line data with control characters onto the spool and use a page definition for post spool formatting, if you specify CTLCHAR(*NONE) and DEVTYPE(*LINE).

Device type considerations

When using line data, you can specify various combinations of DEVTYPE(*LINE), PAGDFN and FORMDF parameter support on the print file. For example:

- Specify DEVTYPE(*LINE), PAGDFN and FORMDF
  - When you print to an AFP printer, PSF/400 uses the PAGDFN and FORMDF parameters to transform the data to IPDS.
  - When you print to a non-AFP printer, the PAGDFN and FORMDF parameters are ignored. The parameters on the print file are used, and the line data is transformed to IPDS or SCS.
- Specify DEVTYPE(*LINE), no PAGDFN, with FORMDF
  - When you print to an AFP printer, an inline page definition is built from the print file parameters. PSF/400 uses the inline page definition and user-specified FORMDF parameter to transform the data to IPDS.
  - When you print to a non-AFP printer, the FORMDF parameter is ignored. The print file parameters are used, and the line data is transformed to IPDS or SCS.
- Specify DEVTYPE(*LINE), PAGDFN, no FORMDF
  - When you print to an AFP printer, an inline form definition is built from the print file parameters. PSF/400 uses the user-specified PAGDFN parameter and the inline form definition to transform the data to IPDS.
  - When you print to a non-AFP printer, the PAGDFN parameter is ignored. The print file parameters are used, and the line data is transformed to IPDS or SCS.
- Specify DEVTYPE(*LINE), no PAGDFN, no FORMDF
  - When you print to an AFP printer, an inline PAGDFN and FORMDF is built from the print file parameters. PSF/400 uses the inline page definition and inline form definition to transform the data to IPDS.
  - When you print to a non-AFP printer, the print file parameters are used, and the line data is transformed to IPDS or SCS.

The support for combinations of PAGDFN and FORMDF for DEVTYPE(*AFPDSLINE) are similar to DEVTYPE(*LINE) line data. However, you should be aware of the following exceptions:

- When you send data to a non-AFP printer, the data can not be transformed to IPDS or to SCS. The spooled file must be printed on an AFP printer.
- Although you are not required to specify PAGDFN or FORMDF with *AFPDSLINE data [3], certain AFPDS commands in the data stream (for example, Invoke Medium Map) may reference named structured fields in the PAGDFN or FORMDF that may not match those in the inline PAGDFN and FORMDF

Carriage control characters

The carriage control character can be represented as either ANSI or machine code.

---

3. You are not required to specify PAGDFN or FORMDF with *AFPDSLINE data because it can be built inline from a print file.

ANSI carriage control is a standard representation that is used with printers from many different manufacturers. Table 18 lists the ANSI codes and their functions. Machine code control characters were defined by IBM; they correspond to channel command words issued by the operating system. Table 19 lists the IBM machine code values and functions.

Note: You may not use both ANSI and machine codes within a single data set.

There are differences in the conventions used by OS/400 for ANSI and machine code line spacing. The OS/400 convention for ANSI handles line spacing and then causes the line to be printed. The OS/400 convention for machine codes causes the line to be printed and then the spacing action is performed.

ANSI carriage control characters

Table 18. ANSI Carriage Control Characters

Control Character Value (in hexadecimal)	Function
X'40' (blank)	Space 1 line, then print (single spacing)
X'F0' (zero)	Space 2 lines, then print (double spacing)
X'60' (dash)	Space 3 lines, then print (triple spacing)
X'4E' (plus sign)	Suppress spacing, then print (overstrike previous line)
X'F1'	Print the data at line position defined as Channel 1 (by convention, the first line on a new page)
X'F2'	Print the data at the line position defined as Channel 2
X'F3'	Print the data at the line position defined as Channel 3
X'F4'	Print the data at the line position defined as Channel 4
X'F5'	Print the data at the line position defined as Channel 5
X'F6'	Print the data at the line position defined as Channel 6
X'F7'	Print the data at the line position defined as Channel 7
X'F8'	Print the data at the line position defined as Channel 8
X'F9'	Print the data at the line position defined as Channel 9
X'C1'	Print the data at the line position defined as Channel 10
X'C2'	Print the data at the line position defined as Channel 11
X'C3'	Print the data at the line position defined as Channel 12
Note: When ANSI carriage controls are used, only the values that appear in this table are considered valid by PSF/400. PSF/400 treats any other ANSI carriage control value as invalid and prints any data on the line using single spacing.	

Machine carriage control characters

Table 19. Machine Code Control Characters

Control Character Value (in hexadecimal)	Function
X'03'	No operation
X'09'	Print and space 1 line (single spacing)
X'11'	Print and space 2 lines (double spacing)
X'19'	Print and space 3 lines (triple spacing)

Table 19. Machine Code Control Characters (continued)

Control Character Value (in hexadecimal)	Function
X'01'	Print without spacing (overstrike next line)
X'89'	Print the data, then skip to the line position defined as Channel 1 (by convention, the first line on a new page)
X'91'	Print the data, then skip to the line position defined as Channel 2
X'99'	Print the data, then skip to the line position defined as Channel 3
X'A1'	Print the data, then skip to the line position defined as Channel 4
X'A9'	Print the data, then skip to the line position defined as Channel 5
X'B1'	Print the data, then skip to the line position defined as Channel 6
X'B9'	Print the data, then skip to the line position defined as Channel 7
X'C1'	Print the data, then skip to the line position defined as Channel 8
X'C9'	Print the data, then skip to the line position defined as Channel 9
X'D1'	Print the data, then skip to the line position defined as Channel 10
X'D9'	Print the data, then skip to the line position defined as Channel 11
X'E1'	Print the data, then skip to the line position defined as Channel 12
X'0B'	Space 1 line without printing
X'13'	Space 2 lines without printing
X'1B'	Space 3 lines without printing
X'8B'	Skip to Channel 1 immediate (by convention, the first line on a new page)
X'93'	Skip to the Channel 2 position immediate
X'9B'	Skip to the Channel 3 position immediate
X'A3'	Skip to the Channel 4 position immediate
X'AB'	Skip to the Channel 5 position immediate
X'B3'	Skip to the Channel 6 position immediate
X'BB'	Skip to the Channel 7 position immediate
X'C3'	Skip to the Channel 8 position immediate
X'CB'	Skip to the Channel 9 position immediate
X'D3'	Skip to the Channel 10 position immediate
X'DB'	Skip to the Channel 11 position immediate
X'E3'	Skip to the Channel 12 position immediate

Table 19. Machine Code Control Characters (continued)

Control Character Value (in hexadecimal)	Function
Note: PSF/400 ignores the following hexadecimal machine-code carriage control characters and does not print lines containing them: X'02' through X'07', X'0A', X'12', X'23', X'43', X'63', X'6B', X'73', X'7B', X'EB', X'F3', and X'FB'. PSF/400 treats any other carriage control value as invalid and prints any data on the line using single spacing.	

Table reference characters (TRC)

Table Reference Characters allow an additional byte to appear at the beginning of a line to indicate which one of up to four different character arrangement tables (coded fonts specified by AFPCHARS parameter) will be used to print the line. This byte, the *table reference character* contains a value of X'F0', X'F1', X'F2', or X'F3', corresponding to the relative position of the desired coded font in the list of coded fonts specified by the AFPCHARS parameter. If carriage control bytes are used with the data, the table reference character follows the carriage control byte but precedes the data bytes. If a carriage control bytes are not used, the table reference character is the first byte of the data record. As with carriage control, if table reference characters are used, every data record must contain a TRC byte.

The following tables summarize the valid forms of line data:

Table 20. Simple data line

D A T A

Table 21. Data line with carriage control byte

CC	D A T A

Table 22. Data line with table reference character

TRC	D A T A

Table 23. Data line with carriage control byte and table reference character

CC	TRC	D A T A

Line data and IGC parameters

The IGC parameters of an OS/400 printer file are described here.

IGCDTA

Indicates IGC data may be used in the file. The user for a line or mixed data file will need to indicate that there is SO/SI present in the data by setting IGCDTA to *YES.

IGCCPI

For AFP printers, this parameter is ignored, as the pitch of the DBCS data is determined by the selected font.

For non-AFP printers, during transformation from line data to SCS, this parameter specifies the pitch of the DBCS data. DBCS SO/SI can not be transformed when going to an IPDS printer.

Mixed data can not be transformed when going to a SCS or IPDS printer.

IGCSOSI

This keyword indicates what action should be taken when SO/SI are found in the data. If the data is mixed, the SO/SI should be taken out and appropriate spaces inserted based on the value of this keyword.

*YES

The SO/SI characters will be printed as blanks.

*NO

The system does not print the shift control characters. These characters do not occupy a position on the printed output.

*RIGHT

The system prints two blanks when printing the shift-in characters but does not print shift-out characters.

IGCEXNCHR

Ignored, as extension character processing only applies to SCS DBCS printer, not AFP attached printers.

IGCCHRRTT

For AFP printers, this parameter is ignored. Character rotation can be specified in the PAGDFN.

For non-AFP printers, when line data is transformed to SCS, this parameter is used to rotate the DBCS data.

The following is a list of DDS keywords that are supported for line data. If you use DDS keywords that are not included on this list while processing line data, they will be ignored.

- ALIAS
- BLKFOLD
- CCSID
- DATE
- DATFMT
- DATSEP
- DFT
- DLTEDT
- INVDTAMAP
- EDTCDE
- EDTWRD
- FLTFIXDEC
- FLTPCN
- IGCALTTYP
- IGCANKCNV
- INDARA
- INDTXT
- INVMMAP
- MSGCON
- PAGNBR
- REF
- REFFLD

- SKIPA
- SKIPB
- SPACEA
- SPACEB
- TEXT
- TIME
- TIMFMT
- TIMSEP

INVDTAMAP (invoke data map) keyword

INVDTAMAP is a record-level keyword to invoke a new data map. It specitfies the name of the data map in a page definition. The page definition is used to map the line data. Data maps in page definitions can perform functions, including multiple-up or rotated printing, changing fonts, and lines per inch. You must have PSF/400 installed to use this keyword.

The format of the keyword is:

```
INVDTAMAP(data-map-name | &data-map-name-field)
```

The data-map-name paramater is required and defines a data map in the page definition. This parameter is 8 characters. The data map name can be specified as a constant or program-to-system field.

This keyword is valid with DEVTYPE(*LINE) or DEVTYPE(*AFPDSLINE). A page definition must be specified on the print file. If DEVTYPE is changed to anything other than *LINE or *AFPDSLINE, the keyword will be ignored and a warning message will be issued at print time.

The INVDTAMAP, SKIP, and SPACE keywords are processed in the following order:

- SKIPB
- SPACEB
- INVDTAMAP
- SPACEA
- SKIPA

The following example shows how to specify the INVDTAMAP keyword.

```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
*
              R RECORD1
        02                                INDTAMAP(MAP1)
              R RECORD2                    INVDTAMAP(&MAP);
                MAP           8A  P

*
```

INVMMAP (medium-map-name) DDS keyword

INVMMAP is a record level keyword in DDS used to invoke a medium map. Invoke Medium Map (IMM) specifies the name of a medium map in a form definition. Use the IMM in the form definition to select or change print parameters such as input drawer, page rotation, overlays.

The medium map name is limited to 8 characters. You can specify the medium map name as a constant or a program-to-system field.
- medium-map-name
- field1

The INVMMAP keyword is valid only with DEVTYPE(*AFPDS). Also, a form definition must be specified on the print file. If DEVTYPE is changed to anything other than *AFPDS, the INVMMAP keyword is ignored and a warning message will be issued at print time.

PSF/400 ends printing on the current sheet when a invoke medium map is encountered.

INVMMAP, SKIP, and SPACE keywords are processed in the following order:
- SKIPB
- SPACEB
- INVMMAP
- SPACEA
- SKIPA

The medium map specified remains in effect for the rest of the file unless changed by another INVMMAP keyword.

The invoke medium map keyword is validated at print time. An error message will be issued if it is not valid.

Option indicators are valid for the INVMMAP keyword.

The following example shows how to specify the INVMMAP keyword.
```
0         1         2         3         4         5         6         7         8
12345678901234567890123456789012345678901234567890123456789012345678901234567890
*
              R RECORD1
       02                                   INVMMAP(MAP1)
              R RECORD2                      INVMMAP(&MAP);
                MAP           8A   P

*
```

If indicator 02 is on, RECORD1 uses a new medium map (MAP1).

RECORD2 allows the application program to specify the name of medium map by setting program variable MAP.

Restrictions when using line data and mixed data

Externally described printer files (DDS SUPPORT) support both line and mixed data device types.

Device type *LINE or *AFPDSLINE and SPOOL(*NO) will not be allowed on the CRTPRTF or CHGPRTF commands. If this condition exists, Message CPD7341, indicating the value SPOOL(*NO) is not valid with device type, will be issued.

Chapter 9. Printing AFPDS data

This chapter discusses two ways of printing AFPDS data:

- Go to "Printing AFPDS data generated on the iSeries server" to read about printing AFPDS data that was generated on the iSeries server.
- Go to "System/390 AFPDS and line data" to read about printing AFPDS and line data generated on a System/390.

Printing AFPDS data generated on the iSeries server

Follow these steps to generate and print OS/400 AFPDS data:

Step 1 Creating AFPDS data.

- You can create AFPDS data on the iSeries server from applications using a printer file with the device type (DEVTYPE) parameter value set to *AFPDS.

- You can create AFPDS data with DDS printing functions that are tailored for use with AFPDS. For more information about the DDS functions that support AFPDS, see the DDS Reference information in the iSeries Information Center.

- You can create and print AFPDS data from an existing database file, merged with an overlay, using the Advanced Function Printing Utilities/400 licensed program.

  You can learn more about this licensed program by going to "Advanced Function Printing Utilities/400" on page 313.

Step 2 Using overlays.

After overlays are created and placed in a library, you can use them by specifying them in the front overlay (FRONTOVL) and back overlay (BACKOVL) parameters of a printer file.

Overlays can also be specified using the DDS OVERLAY keyword. For more information on using overlays in a printer file, go to the DDS Reference information in the iSeries Information Center.

Step 3 Using application program.

The application program that creates the data to be used with the overlays may need to be tailored so that the printed output fits with the overlays that you designed.

Tailoring the program to fit the overlay is similar to tailoring the program to work with a pre-printed form.

System/390 AFPDS and line data

From an advanced function printing (AFP) perspective, data sent to OS/400 from the System/390 can be any of the following:

- Print Data (letters, documents, and so on)
  - Advanced Function Printing Data Stream (AFPDS)
  - AFPDSLINE (AFPDS and line data mixed)

AFPDSLINE is data that has a mixture of line data and AFPDS data. AFPDSLINE contains data that specifies placement and presentation information needed for printing.

- LINE (line data)

  Line data is data that has been prepared for printing on a line printer. Line data does not contain all placement or presentation information that is needed for printing on page printers. Line data is created on the System/390 computer.

- Resources
  - Fonts
  - Form definitions
  - Overlays
  - Page definitions
  - Page segments

Working with the System/390

Communication with the System/390 is important to determine how the System/390 will send data to your iSeries server.

When data is sent from a System/390 to the iSeries server, it can go directly to the output queue of a designated user ID or it can go into your network files.

Notes:

1. Sending print data to the output queue of a user ID is supported only when your host System/390 RSCS/JES subsystem is connected to the iSeries server VM/MVS Bridge.

2. To send data to another system (iSeries server or System/390), you can use the Send Network Spooled File (SNDNETSPLF) command. Go to "Using the send option" on page 143 for more information about this command.

3. The *SNA Distribution Service* guide contains information on how to create communications networks that enable you to send data from the iSeries server to other systems.

If the data consists of print data (AFPDS, LINE, or AFPDSLINE data), the data can be sent directly to an output queue of a specified user ID on OS/400. AFPDS data can go to your network files, but you must then use the Print Advanced Function Printer Data (PRTAFPDTA) command to put the file on an output queue.

If the data consists of resources (fonts, form definitions, overlays, page definitions, or page segments), the data must go into your network files.

Sending print data to the output queue of a user ID

The destination of print data sent from a System/390 to an iSeries server should be the output queue of a specified user ID.

Sending print data from VM System/390 to a user's output queue on OS/400

A VM System/390 can issue either a PRINT command or a PSF command to place a file directly into an OS/400 user's output queue.

Using the PRINT command: The following set of commands places the file named `reports letter a` into the output queue of the user named `userone` on OS/400 with a node ID of `as4002`.

```
'TAG DEV PRT as4002 userone'
'SPOOL PRT TO rscs COPY 1'
'PRINT reports letter a (NOCC'
```

Using the PSF command: The following set of commands places the LINE data spooled file named `reports letter a` into the output queue of the user named `userone` on the iSeries server with a node ID of `as4002`. A PAGEDEF will be used when the data is printed.

When using the PSF command, you must place the characters P1 in front of the PAGEDEF name. When using a FORMDEF, you must place the characters F1 in front of the FORMDEF name. This example uses PAGEDEF; the PAGEDEF object name in the command is P1MYFORM.

```
'TAG DEV PRT as4002 userone'
'SPOOL PRT TO RSCS COPY 1'
'PSF reports letter a (PAGEDEF (P1MYFORM RESOURCE A))'
```

Sending print data from MVS System/390 to a user's output queue on an iSeries server

The following set of commands, when issued by an MVS System/390, places the AFPDS spooled file into the output queue of the specified user on OS/400.

These commands are job control language (JCL) commands. Do not place a P1 (for PAGEDEF) or an F1 (for FORMDEF) in the instructions. In the example below a FORMDEF named MYFORM is used.

```
//INSTR      PROC    NODE='as4002',USER='userone'
                     INFILE='dept265.userx.files(report)'
//SPOOL      EXEC    PGM=IEBGENER
//MYOUT      OUTPUT  DEST=NODE..USER,
                     COPIES=1,
                     FORMDEF=MYFORM
//SYSPRINT   DD      SYSOUT=*
//SYSIN      DD      DUMMY
//SYSUT1     DD      DSN=&,DISP=SHR,DCB=(RECFM=FBA);
//SYSUT2     DD      SYSOUT=A,OUTPUT=*.MYOUT
// PEND
//STEP01     EXEC    PROC=INSTR
```

System/390 parameters and matching OS/400 printer file parameters

Be aware of the following special considerations when working with these System/390 (VM and MVS) parameters:

FCB

If you specify the FCB parameter on VM or MVS, the printer file of that name in the library list of the user receiving the spooled file on OS/400 is used. If the FCB parameter is not specified, the iSeries server printer file QSYSPRT is used.

BIN AND DUPLEX

If the BIN and DUPLEX parameter values are not specified from VM and MVS and the form definition is not specified from VM or MVS, the values for the DRAWER and DUPLEX parameters are taken from the printer file specified in

the FCB parameter. If the FCB parameter is not specified, the DRAWER and DUPLEX values are set to *FORMDEF, meaning that the values are taken from the form definition.

FORMDEF

If the form definition (FORMDEF) parameter is not specified from VM or MVS, the OS/400 form definition parameter value is set to *DEVD and *LIBL is used for the library. *DEVD means the form definition used is the one specified in the device description for the printer you want to use.

The following System/390 parameters are supported when data is sent directly to the output queue of a user on the iSeries server.

If a System/390 printer parameter that is not in this table is specified, and there is no matching or equivalent OS/400 printer file parameter, the System/390 printer parameter is ignored when the data is printed on the iSeries server.

Table 24. VM System/390 Parameters

System/390 Printer Parameter Names	VM Commands	Function	OS/400 Printer File Parameter Names
BIN	PSF	Specifies which drawer or bin the paper is taken from.	DRAWER
CC	PRINT, PSF	Specifies control characters.	CTLCHAR[1]
CHARS	SPOOL	Specifies a table of coded fonts.	AFPCHARS
COPY	SPOOL	Specifies the number of copies.	COPIES
DATACK	PSF	Specifies whether or not the printer will block print positioning and invalid character errors. Twinaxial attached printers will always block print positioning and invalid character errors unless the printer file used to create the spooled file has the fidelity parameter value set to *ABSOLUTE.	No equivalent OS/400 parameter
DEST	TAG	Specifies node and user ID.	No equivalent OS/400 parameter
DUPLEX	PSF	Specifies if duplex printing is to be used.	DUPLEX
FCB	SPOOL, TAG	Specifies the printer file used.	FILE
FORM	SPOOL	Specifies the form type to be used.	FORMTYPE
FORMDEF	PSF	Specifies the form definition to be used.	FORMDF
PAGEDEF	PSF	Specifies the page definition to be used.	PAGDFN
PRMODE[2]	PSF	Specifies the device type, ideographic character data, processing shift-out/shift-in characters.	DEVTYPE, IGCDTA, IGCSOSI
TRC	PRINT, PSF	Specifies if the data stream contains table reference codes.	TBLREFCHR

Table 24. VM System/390 Parameters (continued)

System/390 Printer Parameter Names	VM Commands	Function	OS/400 Printer File Parameter Names
[1] In order to correctly print System/390 line data with first-character forms control, each channel value specified in the CTLCHAR parameter must have a unique line number associated with that channel value. [2] Only PRMODE values of PAGE, LINE, SOSI1, and SOSI2 are supported. The supported PRMODE values translate to device types of *AFPDS, *LINE, or *AFPDSLINE. See Network Job Entry Formats and Protocols for more specific information			

Table 25. MVS System/390 Parameters

System/390 Printer Parameter Names	MVS JCL Statements	Function	OS/400 Printer File Parameter Names
CHARS	DD, OUTPUT	Specifies a table of coded fonts.	AFPCHARS
CONTROL	OUTPUT	Specifies line spacing.	No equivalent OS/400 parameter
COPIES	DD, OUTPUT	Specifies the number of copies.	COPIES
DATACK	OUTPUT	Specifies whether or not the printer will block print positioning and invalid character errors. Twinaxial attached printers will always block print positioning and invalid character errors unless the printer file used to create the spooled file has the fidelity parameter value set to *ABSOLUTE.	No equivalent OS/400 parameter
DCB=RECFM	DD	Specifies control characters.	CTLCHAR[1]
DEST	DD, OUTPUT	Specifies node and user ID.	No equivalent OS/400 parameter
FCB	DD, OUTPUT	Specifies the printer file used.	FILE
FORMDEF	OUTPUT	Specifies the form definition to be used.	FORMDF
FORMS	OUTPUT	Specifies the form type to be used.	FORMTYPE
PAGEDEF	OUTPUT	Specifies the page definition to be used.	PAGDFN
PRMODE[2]	OUTPUT	Specifies the device type, ideographic character data, processing shift-out/shift-in characters.	DEVTYPE, IGCDTA, IGCSOSI
TRC	OUTPUT	Specifies if the data stream contains table reference codes.	TBLREFCHR

System/390 Printer Parameter Names	MVS JCL Statements	Function	OS/400 Printer File Parameter Names
[1] In order to correctly print System/390 line data with first-character forms control, each channel value specified in the CTLCHAR parameter must have a unique line number associated with that channel value. [2] Only PRMODE values of PAGE, LINE, SOSI1, and SOSI2 are supported. The supported PRMODE values translate to device types of *AFPDS, *LINE, or *AFPDSLINE. See Network Job Entry Formats and Protocols for more specific information			

Managing print data sent to an OS/400 output queue

The ability of the System/390 to send these files directly to the output queue of a specified user ID on the iSeries server eliminates the steps of receiving the files to a library and then using the PRTAFPDTA command to print them. The following diagram shows the path of the files from the System/390 to the OS/400 printer.

Note: If you cannot find the spooled file on the output queue of the specified user ID, look in the QNFTP job log for informational, diagnostic, and error messages. The QNFTP job runs under the QSNADS user profile.

RV3H050-0
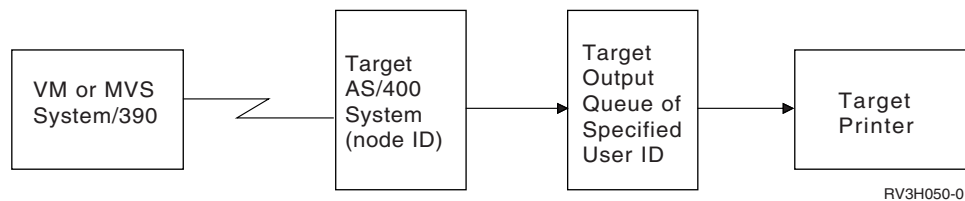
Target iSeries server

The target iSeries server is the system in a communications network that the System/390 is sending the files to. The iSeries server is providing the function of a file server for the host System/390. The iSeries server is known to the host System/390 by a unique node name (node ID).

Note: The target user ID must have access to the libraries containing the resources (overlays, fonts, page segments) that the spooled file needs when it prints on the iSeries server. Go to "Working with AFP resources and libraries" on page 202 for more information about resources and libraries.

Target user ID

The host System/390, working with the iSeries server, can send spooled files directly to the output queue of a specified user ID. This capability eliminates manually receiving spooled files and then using the PRTAFPDTA command to print the files.

Target output queue

Caution should be used when having the sent file placed directly on the output queue. Spooled files placed directly into the output queue of the user ID could be printed without the owner of the user ID being aware of it if a printer is active for that output queue.

Possible solutions are:

- Create a special user ID and a special output queue for that user ID.

  You could then sign on using that user ID and assign a printer to your output queue to print the spooled files.
- Make sure the output queue of the receiving user ID is held before the System/390 sends the files.
- Use the End Writer (ENDWTR) command to end the printer assigned to that output queue before the System/390 sends the files.

Target printer

Make sure that the printer you want to print the files on is assigned to the correct output queue. If any special handling of the printer is necessary (form choice, drawer selection, envelopes, and so on), it should be done before the files are sent.

Sending resources and AFPDS data to network files

The destination of advanced function printing resources (fonts, overlays, page segments, page definitions, and form definitions) sent from a System/390 to an iSeries server must be network files.

Note: AFPDS data can also be sent to network files. However, this is not the recommended way to send this data to an iSeries server. If you choose this way, you must use the Print Advanced Function Printer Data (PRTAFPDTA) command to put the file on an output queue.

Sending resources and AFPDS data from VM System/390 to network files

The following command, when issued by a VM System/390, places the specified file into the network files of a user on an iSeries server.

```
SENDFILE fonts resource a TO userone AT as4002
```

Sending resources and AFPDS data from MVS System/390 to network files

The following command, when issued by an MVS System/390, places the specified file into the network files of a user on an iSeries server.

```
TRANSMIT as4002.userone DSNAME('system.afp.resources(font)')
```

Receiving resources and AFPDS data sent to network files

To make receiving resources and AFPDS data as easy as possible, you should be aware of the following:

- Make sure you do not send LINE data or mixed data (AFPDSLINE data) to network files.
- The AFPDS files, as they exist on System/390, are in variable-length record format. Make sure the System/390 site converts the AFPDS files to a fixed length record format. If this is not done prior to sending the AFPDS files, large amounts of iSeries server disk space will be required. You need to reach an agreement with the System/390 site as to what fixed-length record format you will use; for example, 4000 bytes. AFPDS records longer than the fixed-length record size must be converted into multiple fixed-length records, with the last record padded out with blank characters (if necessary).
- Before receiving AFPDS files from the System/390, you should consider creating a fixed-length physical data file to hold the files from the System/390. The command to do this is the CRTPF (Create Physical File) command.

  The length of the records in the physical data file should be the number that you and the System/390 site agreed to; for example, 4000 bytes.

Many types of resources are sent to the iSeries server. Administratively, it is more convenient to store the different types of resources in separate physical files. That is, put all the font data in a physical file that contains only fonts, the overlay data in a physical file that contains only overlays, and so on.

The following sample command creates a physical file that can receive resources and AFPDS data:

```
CRTPF FILE(MYLIB/MYFILE) RCDLEN(4000) MBR(*NONE) MAXMBRS(*NOMAX) LVLCHK(*NO)
```

Using the Work with Network Files (WRKNETF) and Receive Network File (RCVNETF) Commands

These commands can be used to receive resources and AFPDS files from the System/390.

To see a prompt display for these commands, type WRKNETF or RCVNETF and press F4 (Prompt).

The Work with Network Files (WRKNETF) command allows you to work with a list of files that have arrived for a user, or creates a database file containing a list of the files.

If the list is displayed, you can enter an option to select a function to be performed on the file. You can:

- Receive the file into a user file.
- Delete the file.
- Browse the file (not valid for save files).
- Submit files (submit the job).

Restrictions: (1) A user with security officer authority can display the network files for any user. If you are not the security officer, you can display only those files that were sent to you or to your group profile. (2) To perform any of the options from this display, you must be authorized to the command corresponding to that option. For example, you must be authorized to the Browse Physical File Member (BRWPFM) command for the browse function, and the Submit Database Jobs (SBMDBJOB) command for the submit job function.

Examples

The following command allows you to work with all network files for the user running this command.

```
WRKNETF
```

If the command is issued as an interactive job, the list of files is displayed at the requesting work station. If the command is issued as a batch job, the list of files is printed with the job's spooled output.

The following command allows you to work with the network files for USR1 and prints the output with the job's spooled output.

```
WRKNETF  USER(USR1)  OUTPUT(*PRINT)
```

This command can only be issued by USR1, a member of the USR1 group, or a user with security officer rights.

The following command allows you to work with the network files for all users and is written to the first member of a database file named NETFILES.

```
WRKNETF  USER(*ALL)  OUTPUT(*OUTFILE)  OUTFILE(NETFILES)
```

If the file exists in a library on the library list, the existing file is used; otherwise, the file is created in library QGPL. If the file did not exist, or did not contain any members, a member with the same name as the file is added to the file; otherwise, the first member of the file is cleared and used. This command can be issued only by a user with security officer rights.

The following command receives the network file SCRIPT, member $REPORT, into a physical file named MYFILE in library MYLIB.

```
RCVNETF FROMFILE(SCRIPT) TOFILE(MYLIB/MYFILE) FROMMBR($REPORT)
```

The new member in MYFILE is $REPORT.

Creating resources on the iSeries server

When resource data is received from System/390, use the following commands to convert the resources to a format that can be used by OS/400.
- CRTPAGSEG
- CRTOVL
- CRTFORMDF
- CRTFNTRSC
- CRTPAGDFN

Printing AFPDS data on the iSeries server

The PRTAFPDTA command can be used to print AFPDS files.

If you want to use a prompt display with this command, type PRTAFPDTA and press F4 (Prompt).

Notes:
1. The PRTAFPDTA command prints AFPDS data. Any padding between the AFPDS structured fields must be hex 40. The padding cannot be hex 00.
2. When using the PRTAFPDTA command, do not override the DEVTYPE parameter on the QSYSPRT printer file.
3. When using the PRTAFPDTA command, the form definition usually identifies the drawer to print from. An exception is when an Override Printer File (OVRPRTF) command has been issued with a different drawer value. In this case, the drawer value is taken from the override command.

Examples
The following command prints the first member in file MYFILE starting with page 2 and ending on page 6.

```
PRTAFPDTA  FILE(MYLIB/MYFILE) STRPAGE(2)  ENDPAGE(6)
```

The following command prints the member $REPORT in file MYFILE using a form definition of F10101 and all available exception handling.

```
PRTAFPDTA  FILE(MYLIB/MYFILE) MBR($REPORT)  FORMDF(F10101)  FIDELITY(*CONTENT)
```

Additional PSF/400 functions
PSF/400 provides functions that are accessible through a mechanism separate from DDS, printer files, printer device descriptions, or the Start Printer Writer (STRPRTWTR) command. These additional functions allow you to specify how PSF/400 works with printers and fonts.

Chapter 10. Working with print services facility (PSF) configuration objects

This chapter provides information about PSF configuration objects and some of the print functions that are enabled by them:

- IPDS pass-through
- IPDS to PDF transform
- Session and IPDS dialog sharing
- User resource library list
- APPC and TCP/IP Retry Count (RETRY)

About PSF configuration commands

There are several PSF configuration commands you can use to create, change, delete, display, and work with a PSF configuration object. The following provides a brief description about each command. For information about specific parameters of the configuration object, see online help.

Creating a PSF configuration object

To create a PSF configuration object, use the Create PSF Configuration (CRTPSFCFG) command. This command allows you to specify additional parameters for an AFP printer that are not supported in the Create Device Description (Printer)(CRTDEVPRT).

Changing a PSF configuration object

To change a PSF configuration object, use the Change Print Services Facility Configuration (CHGPSFCFG) command. This command changes the PSF configuration object in the library specified on the command.

Displaying a PSF configuration object

To display a PSF configuration object, use the Display Print Services Facility Configuration (DSPPSFCFG) command. This command allows the you to display or print the contents of a PSF configuration object.

Deleting a PSF configuration object

To delete a PSF configuration object, use the Delete Print Services Facility Configuration (DLTPSFCFG) command. This command deletes a PSF configuration object form a specified library. If the PSF configuration is found, it is deleted. If it is not found, a message is sent to the user indicating that the PSF configuration object could not be found.

Working with PSF configuration objects

To work with a PSF configuration object, use the Work with PSF configuration (WRKPSFCFG) command. This command allows the user to work with all of the PSF configuration objects from the system or user libraries.

Using PSF configuration objects

To use a PSF configuration object, name the object with the USRDFNOBJ parameter and specifying AFP(*YES) on either the CRTDEVPRT or CHGDEVPRT commands.

Working with IPDS pass-through support for PSF for OS/400

This topic provides information about Intelligent Printer Data Stream™ (IPDS) pass-through support and how to enable it to work with printers attached to the iSeries server. IPDS pass-through support is a function that allows Print Services Facility for OS/400 (PSF/400) to accept print data in the IPDS format and pass the data directly to a printer. Print data in this format that is directed to a supported printer is no longer transformed into AFPDS format before it is made available to PSF/400.

IPDS pass-through support is available for any IPDS printer that provides resident fonts. IPDS pass-through support is not available for the following IBM IPDS printers: 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900-001 and any printer attached using the DPF function provided by Print Services Facility for OS/2. These printers do not provide resident fonts.

Why use IPDS pass-through support?

You may want to use IPDS pass-through support if you have applications that generate SCS or IPDS data streams when printing to an AFP printer. Consider using IPDS pass-through if the following apply:

- You are using applications such as Business Graphics Utilities, GDDM®, or Virtual Print that does not support AFPDS.
- Your SCS or IPDS application does not contain any reference to overlay page segments or host font character sets.

How IPDS pass-through function works

The following figure shows the data stream origination, flow, and conversion for a spooled file when IPDS pass-through support is enabled.

AS/400 System with IPDS Pass-through support enabled

Application

DDS specifications   Print options

Spooled file in output queue

Writer

(AFP data stream)   (IPDS data stream)   (SCS data stream)

(IPDS data stream)   (IPDS data stream)

AS/400

Device

AFP printer

RBAFT525-0

IPDS pass-through limitations

Consider the following IPDS pass-through limitations which determine if a job or printer is not eligible for IPDS pass-through:

- Some SCS or IPDS jobs are not eligible for IPDS pass-through because they contain special functions that require transform to AFPDS for the job to print correctly. Specifying IPDS pass-through in the PSF configuration object allows only those jobs eligible for IPDS pass-through to bypass the extra transforms. Those jobs not eligible for IPDS pass-through are still transformed to AFPDS and back to IPDS.

  Spooled files that contain the following data or printer file attributes are not eligible for IPDS pass-through:

  – SCS or IPDS data streams that contain image object content architecture (IOCA) or IM1

  – For SCS or IPDS data stream, a front or back overlay was specified in the printer file.

  – For SCS data stream, IGCDTA(*YES) was specified in the printer file.

  – For SCS or IPDS data streams, MULTIUP with REDUCE(*NONE) was specified in the printer file.

  – For SCS or IPDS data stream, a Form definition was specified in the printer file.

– When specifying EDGESTITCH or CORNERSTPL finishing operations in the printer file.

If IPDS pass-through is specified in the PSF configuration object and the job is not eligible for IPDS pass-through, a diagnostic message is sent and the print file is transformed to AFPDS and then back to IPDS.

- Not all printers supported by PSF for OS/400 can use IPDS pass-through. This is because the resident fonts referenced in the data stream must be mapped to the host fonts which are downloaded to the printer. The following printers can not support IPDS pass-through when AFP(*YES) is specified: 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900-001 and any printer attached using the DPF function provided by Print Services Facility for OS/2.
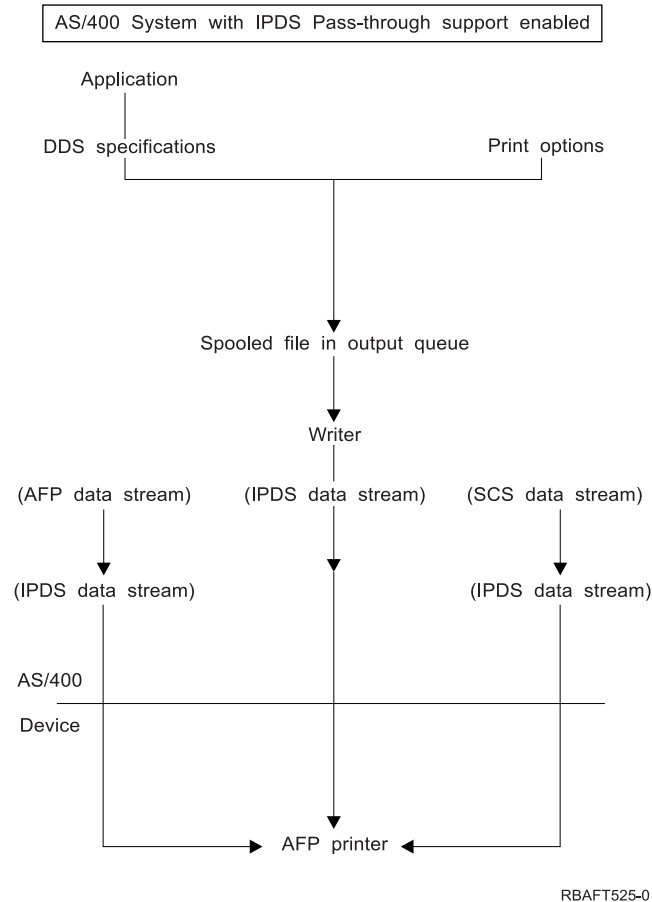
When a printer writer is started to a printer configured with AFP(*YES) that does not support IPDS pass-through and IPDSPASTHR(*YES) is specified, a diagnostic message is sent stating that IPDS pass-through is not supported. The printer file is transformed to AFPDS and then back to IPDS.

Enabling IPDS pass-through support

IPDS pass-through support is enabled by doing the following:

- You can enable IPDSPASTHR by specifying *YES to the IPDSPASTHR parameter in the printer file. You can also enable IPDSPASTHR by specifying *DEVD for the IPDSPASTHR parameter in the printer file and then specifying IPDSPASTHR in the PSF configuration object.

Parameters supporting IPDS pass-through

The following PSF configuration object parameters are used by the IPDS pass-through function:

PSFCFG
    Print services facility configuration object. The possible values are:

    PSF configuration name
            Specify the name of the PSF configuration object being created. The name of the PSF configuration object is qualified by one of the following.

    *CURLIB.
            The current library for the job is used for creating the PSF configuration object. If no library is specified as the current library for the job, library QGPL is used.

    Library-name
            Specify the name of the library where the PSF configuration object is being created.

IPDSPASTHR
    IPDS Pass-through

    *NO   No IPDS pass-through is done.

    *YES. IPDS pass-through is done for the device for all spooled files that are eligible for IPDS pass-through.

Printer device description: The following CRTDEVPRT parameter can be used to enable IPDSPASTHR.

USRDFNOBJ
    User defined object.

*NONE.
No user-defined object is used.

Element 1: Name of User-Defined Object
The name of the user-defined object is qualified by one of the following.

- *LIBL: All libraries in the library list for the job are searched until the first match is found.
- *CURLIB: The current library for the job is searched. If no library is specified as the current library for the job, library QGPL is used.
- library-name: Specify the name of the library to be searched.
- Object-name: Specify the user-defined object to be used by the user application or the user specified program that processes the spooled files.

Element 2: Type of User-Defined Object

The possible values are:

- PSF configuration object (*PSFCFG)

Note: On this element, the only valid value for IPDS pass-through is *PSFCFG.

Printer file: The following printer file parameters can be used to enable IPDSPASTHR support:

IPDSPASTHR
IPDS Pass-through, which allows you to specify IPDSPASTHR on a file-by-file basis.

*DEVD
This is the default value. When specified, PSF/400 will first check for IPDSPASTHR in USRDFNDTA. PSF/400 will then check for a value for IPDSPASTHR in the PSF configuration object that is specified in the printer device description, if one is specified. If there is no object that is specified in the printer device description, IPDS pass-through is ignored.

*NO No IPDS pass-through. This value takes precedence over a value that is specified for IPDSPASTHR in a PSF configuration object.

*YES Specifying *YES for a printer file enables IPDS pass-through for the spooled file. This value takes precedence over the value that is specified for USRDFNDTA in a printer file.

USRDFNDTA
User defined data. These values were available prior to V4R3, allowing you to specify IPDSPASTHR on a file-by-file basis. However, now they are used by PSF/400 for migration purposes, and only if you specify *DEVD for the new IPDSPASTHR parameter. These values take precedence over a value that is specified for IPDSPASTHR in a PSF configuration object.

IPDSPASTHR(*NO)
No IPDS pass-through is done. Note that this is the same as not specifying IPDSPASTHR in the USRDFNDTA parameter.

IPDSPASTHR(*YES)
IPDS pass-through is done for the spooled file if it is eligible for IPDS pass-through.

IPDS to PDF transform

This topic provides information about IPDS to PDF transform support. The IPDS to PDF transform support allows the creation of PDF files from AFP output files. You can use the PDF files in electronic mail applications, internet-based softcopy repositories, or for printing to PDF printers. The PSF transform interface supports an IPDS to PDF transform program if it emulates a TCP/IP-connected IPDS 4028, 3812, or Infoprint 40 printer device.

PSF can mail the PDF files electronically by specifying the PDFGEN *MAIL parameter value (see "CRTPSFCFG parameters" on page 241). After the PDF file creation, PSF/400 retrieves a mail tag from the USRDFNDTA or the STRPAGGRP DDS keyword. You can use Using CRTPSFCFG PDFMAPPGM to specify a user exit program to map the mail tag into one or more email addresses and optional message text. If you do not specify an exit program, you must provide a valid email address in the USRDFNDTA parameter of the printer file or in the STRPAGGRP DDS keyword. PSF/400 then electronically mails the output by using the SNDDST command.

You can handle the distribution function manually by routing the PDF output to a directory by using CRTPSFCFG PDFGEN and PDFDIR. You can route the PDF output to a PDF printer by using the CRTPSFCFG PDFGEN *SPLF paramater value with the PDFOUTQ paramater, .

To subdivide an output file into multiple PDF files, you can use the CRTPSFCFG PDFMULT keyword with the DDS keywords STRPAGGRP and ENDPAGGRP. PSF can mail each to a different address. For example, you can subdivide a customer statement run on customer statement boundaries, by using the DDS STRPAGGRP and ENDPAGGRP keywords. . The STRPAGGRP for each customer statement can specify a mail tag. The mail tag can be, for example, a customer number, that a user exit program can convert to an email address. For each customer statement, the user exit program can instruct PSF either to mail, or not mail, the output to one or more addresses.

An IPDS to PDF transform that can be used with PSF/400, is provided in the Infoprint Server for AS/400 product. The transform converts IPDS controls and data into PDF files. The files include with text, graphics, and image data. The transform also stores the mail tag from USRDFNDTA or STRPAGGRP in a comment statement in the PDF output. For detailed information on the IPDS to PDF transform (including the user exit), see the Infoprint Server for AS/400 User's Guide (Sxxx-xxxx).

Format of the printer file's USRDFNDTA mail information

The format of the mail information when specified in USRDFNDTA is:

```
USRDFNDTA('MAILTAG(80 bytes of text) MAILSENDER(10 byte name)')
```

The 80 bytes of text can contain a mail address or a tag that is converted by the mail exit routine into a mail address and optional message. The MAILSENDER parameter is 10-bytes long and specifies the name of the user profile that is sending the file. This parameter overrides CRTPSFCFG PDFSENDER.

Format of the STRPAGGRP mail tag

When specifying PDFMULT (*YES), the name parameter of STRPAGGRP must contain a mail tag. The mail tag can contain either a mail address or a tag that is

converted by the user exit routine into a mail address and optional message. The
STRPAGGRP name overrides a mail tag in USRDFNDTA

IPDS to PDF transform device configuration

You can configure the IPDS to PDF transform in the same way as a TCP/IP IPDS
printer. The only difference is that the remote location address must be a loopback
address of 127.0.0.1. Use CRTDEVPRT to configure the printer and CRTPSFCFG to
specify PSF configuration parameters.

CRTDEVPRT parameters

You should specify the following CRTDEVPRT parameter values when configuring
a TCP/IP attached printer that will convert IPDS to PDF:

- Device description: printer-name
- Device class: *LAN
- Device type: *IPDS
- Device model: 0
- LAN attachment: *IP
- Advance function printing: *YES
- Port number: xxxx (unique number for each printer)
- Font: printer font
- Remote location: 127.0.0.1
- User defined object: PSF configuration object

You can have multiple IPDS to PDF printers active, but you must configure each
printer with a different port.

CRTPSFCFG parameters

You should specify the following CRTPSFCFG parameter values when configuring
a TCP/IP attached printer that will convert IPDS to PDF. See the CRTPSFCFG
command description in the iSeries Information Center for details on the options
you can specify on these parameters. See also the InfoPrint Server for iSeries:
User's Guide in the iSeries Information Center Supplemental Manuals category for
more information.

PDFGEN
> Specifies whether the user would like the spooled file converted to PDF
> using the IPDS-to-PDF transform. It also specifies what to do with the
> associated PDF output.

PDFDEVTYPE
> Specifies the type of device that the IPDS to PDF transform virtual printer
> device should emulate.

PDFPPRDWR1 (drawer 1), PDFPPRDWR2 (drawer 2)
> Specifies the paper size to use, for either drawer 1 or drawer 2, during the
> IPDS to PDF transform process.

PDFMULT
> Specifies whether to split the PDF output into multiple files by using the
> DDS keyword STRPAGGRP boundaries.

PDFINCFNT
> Specifies whether the PDF output generated by the IPDS to PDF transform
> should carry the necessary fonts inline.

| PDFDTAQ
| Specifies the name and library of the data queue where PSF will log the
| IPDS to PDF transformation completion notifications.

| PDFMAILSVR
| Specifies what mail server(s) to use for e-mailing the resulting PDF files
| from the IPDS to PDF transform.

| PDFSENDER
| Specifies the owner, or sender, of the electronically-mailed PDF output file.

| PDFMAPPGM
| Specifies the qualified name of a user exit program.

| PDFOUTQ
| Specifies the qualified name of an output queue to use when spooling PDF
| output.

| PDFDIR
| Specifies the directory in which to store the PDF file.

Sharing print sessions and IPDS dialogs

Several parameters from the PSF configuration object allow you to configure
PSF/400 to share sessions with other PSFs or ASCII print drivers automatically.

Parameters supporting printer session and dialog sharing

Described below are descriptions of the parameters you need to consider for
printer session and dialog sharing.

ACTRLSTMR: Specifies the point at which the Release Timer (RLSTMR) is
activated. Valid values are *NORDYF, *IMMED, *PRTNORDYF and *PRTIMMED.

*NORDYF
    Specifies that the Release Timer (RLSTMR) is activated when there are no
    more spooled files to print and all pages of the last spooled file have been
    printed. If the RLSTMR expires, the session attached to the printer is
    released. When the session is released, another PSF, such as PSF/400,
    PSF/MVS, PSF/VM, PSF/VSE, PSF for OS/2 or PSF for AIX may start a
    session to the printer.

    If a spooled file becomes ready after the Release Timer has been activated
    and not yet elapsed, use of the timer is stopped. When there are no more
    spooled files to print and all pages of the last spooled file have been
    printed, the timer is activated again using its full specified value,

    PSF will attempt to start a session with the printer when the value
    specified with RESTRTMR expires and there is a spooled file with a status
    of RDY.

    Use this parameter when you are sharing either a single TCP/IP port such
    as an Infoprint printer, or a single APPC remote location (PSF Direct) with
    another PSF.

*IMMED
    Specifies that the Release Timer is activated immediately after PSF has
    successfully started a session with the printer or PSF Direct. If the RLSTMR
    expires, the session to the printer is released. If a spooled file is being sent
    to the printer and the timer elapses, the session is released after all pages
    of the spooled file have been printed.

PSF will attempt to start a session with the printer when the value specified with RESTRTMR expires and there is a spooled file with a status of RDY.

Use this parameter when you are sharing either a single TCP/IP port such as an Infoprint printer, or a single APPC remote location (PSF Direct) with another PSF.

*PRTNORDYF

Specifies that the Release Timer (RLSTMR) is activated when:

- The printer has indicated that it has non-IPDS files to print,
- There are no more spooled files for PSF to print, and
- All pages of the last spooled file have been printed.

After the RLSTMR expires, the IPDS dialog with the printer is released. When the dialog is released, the printer may print non-IPDS files queued at the printer.

If a spooled file becomes ready after the Release Timer has been activated and not yet elapsed, use of the timer is stopped. When there are no more spooled files to print and all pages of the last spooled file have been printed, the timer is activated again using its full specified value,

PSF does not release the session when the dialog is released.

PSF will start a dialog with the printer when the value specified with RESTRTMR expires and there is a spooled file with a status of RDY.

This value is ignored if the printer does not support dialog sharing. PSF does not release the session until the writer is ended.

When PSF is in between spooled files and has not released a dialog, PSF queries the printer every 60 seconds to see if the printer has a non-IPDS file to be printed.

Use this parameter when you are sharing IPDS and non-IPDS printing on a printer that supports IPDS dialog management.

*PRTIMMED

Specifies that the Release Timer (RLSTMR) is activated as soon as the printer has indicated that it has non-IPDS files to print. The IPDS dialog with the printer is released when the RLSTMR has expired. If the timer expires while PSF is sending a spooled file to the printer, PSF does not release the dialog until the spooled file has been completely printed. PSF will then release the dialog even when there are spooled files with a status of RDY on the output queue in use by the writer.

PSF does not release the session when the dialog is released.

When the dialog is released, the printer may print non-IPDS files queued at the printer.

PSF will start a dialog with the printer when the value specified with RESTRTMR expires and there is a spooled file with a status of RDY.

When PSF is in between spooled files and has not released a dialog, PSF queries the printer every 60 seconds to see if the printer has a non-IPDS file to be printed.

Use this parameter when you are sharing IPDS and non-IPDS printing on a printer that supports IPDS dialog management.

If you are using a non-impact continuous forms printer, do not set PRTRQSTMR in the printer device description to *NOMAX. If you set the PRTRQSTMR to *NOMAX, an operator must perform a manual NPRO at the printer. PSF will not release a session, or IPDS dialog, until printing is complete for the last file that is sent to the printer.

ACTTMR: Specifies the length of time (in seconds) that PSF/400 waits for a TCP/IP-attached printer to respond to an activation request. PSF uses this timer only after a session has been opened with the printer.

No timers can influence the length of time that TCP/IP takes to wait for a response to a session start request. But if you want PSF to retry session start requests after a failure of the session start request failure is detected, you may use PSF configuration object parameter RETRY to specify how many session start request retries PSF should perform.

ACTTMR is specified in the printer device description. In previous releases, ACTTMR was specified differently.

RESTRTMR: Specifies the amount of time to wait before the printer writer attempts to reestablish either a session or dialog. For example, if this value is set to 10, PSF/400 activates the timer as soon as a session or dialog is released. After 10 minutes have elapsed, PSF/400 checks for a spooled file with a status of RDY. If there is one, PSF/400 attempts to start a session or dialog with the printer in order to print the file. If a spooled file is not ready after the timer expires, PSF/400 will wait for a spooled file with a RDY status to be placed on its output queue before attempting to start a session or dialog.

This timer is ignored if PSF/400 has not been configured to release either a session or dialog. Use the ACTRLSTMR and RLSTMR parameters to configure PSF/400 to release either a session or dialog.

RETRY: Specifies the number of additional attempts PSF will make to establish an APPC or TCP/IP session. When PSF/400 is sharing a PSF directly attached printer with other PSFs, you may want to set the value to *NOMAX so that PSF/400 will continue retrying to obtain a session. The default value for the RETRY parameter has been changed from 2 to 15.

The RETRY parameter is helpful for TCP/IP-attached printers in the following situations: You wish to share an AFCCU printer with more than three PSFs. A printer is temporarily unavailable when PSF attempts to open a session with it. You may have seen this when PSF failed to start a printer session. One example of this is when PSF issues message PQT3603 with error code 22. Use of RETRY specifies that PSF is to not terminate and continue retrying, up to the number of times specified, to open a session with the printer.

RETRYDLY: Specifies the number of seconds to wait between each retry attempt to establish a session. This parameter is used only when PSF/400 is using an SNA attachment. The default value for the RETRYDLY parameter has been changed from 0 to 90.

RLSTMR: This is the timer whose value is referenced by the ACTRLSTMR parameter. The default setting is *NOMAX, which means that PSF/400 does not release the session or dialog with the printer until the writer is ended.

The value specified for parameter RESTRTMR determines when PSF/400 reestablishes a session or dialog.

Additional information on session sharing

1. PSF/400 supports session sharing with TCP/IP-attached printers.
2. PSF/400 supports session sharing with any printer attached to PSF Direct.
3. PSF/400 does not support session sharing with APPC-attached printers. If you wish to use session sharing with these printers, either use their TCP/IP support, if available, or attach them to PSF Direct, if supported.
4. PSF/400 does not support session sharing with Twinaxial-printers attached to local or remote workstation controllers. If you want to use session sharing with these printers, check if TCP/IP support is available.

Parameters supporting automatic session recovery

The following parameters support automatic session recovery:

AUTOSSNRCY
Automatic session recovery. This specifies whether PSF/400 will automatically attempt to resume printing when a session has been unexpectedly ended by a printer or other network device, such as a router. The acknowledgement request responses from the printer contain information that indicates the pages sent to the printer. AUTOSSNRCY(*YES) also lets you power your printer off and on without ending PSF. If you plan to use a value of *YES, you should consider the following values set for other parameters:

- For a TCP/IP connection, use RETRY or ACKFRQ in the PSF configuration object.
- For an APPC connection, use RETRY, RETRYDLY or ACKFRQ in the PSF configuration object.

Note: When automatic session recovery takes place, pages may be printed when printing resumes.

ACKFRQ
Acknowledgment frequency. It specifies the frequency, in pages, with which PSF/400 sends IPDS acknowledgment requests to a printer. The acknowledgment request responses from the printer that contains information as to the status of pages that is sent to the printer. You may wish to consider adjusting this value if:

- you are concerned about losing a connection to a printer while spooled files are being printed.
- you are concerned about duplicate pages being printed.

More frequent acknowledgment request responses provide more information about pages that have been printed. If PSF is configured to reconnect to a printer when the session is ended abnormally, and PSF is in the middle of printing a spooled file, the number of pages reprinted is minimized but not eliminated; the nature of an abnormal session termination is such that eliminating the reprinting of pages is not possible. If you decide to increase the frequency of acknowledgment requests (by specifying a lower value), the printing process can be slowed down, especially

if a very low value is specified. You may need to experiment with different values in order to find one that is suitable for your environment.

User and device resource library lists

You may specify a User and Device Resource Library List in the PSF configuration object. These libraries will be used to searched for AFP resources needed to print the spool file. The user library list allows you to specify what user libraries are to be searched for their jobs. It may vary from user to user while the device resource library list will be the same for the device. The user library list is searched first and then the device library list.

User resource library list

You can specify libraries to search for font and AFP resources in the user resource library list. The value specified in this paramater will only be used when the USRRSCLIBL parameter of the printer file has a value of *DEVD.

Values supported for the User Resource library list are:

*JOBLIBL
Specifies that the job's current library list that created the spool file is used in searching for AFP resources. The job's library list at the time the spool file is created is saved. Each time you create a new spool file, the current job library list is saved. This is the default.

*CURLIB
The current library for the job that created the spool file is used for searching for AFP resources. If no library is specified as the current library for the job, then library QGPL is used.

*NONE
No user resource library list will be used for searching for AFP resources. Only the device resource library list will be used.

Device resource library list

The device resource library list allows you to specify a device resource library list to be used for searching AFP resources. The user resource library list is searched first and then the device resource library list is searched when attempting to find an AFP resource specified with the spool file.

The following are valid values for the device resource library list:

*DFT Specifies that the following libraries (if present on system) will be used in searching for AFP resources.
- QFNTCPL
- QFNT01 - QFNT19
- QFNT61 - QFNT69

device-resource-library-names
This is a list of up to 30 library names that will be used to search for AFP resources.

The value in using the user resource library and device resource library lists is two-fold. It enhances usability since you can now specify one set of libraries for a 240 pel printer and another set for a 300 pel printer. You can also make sure that the libraries that contain the resources needed by this printer job are actually in the

library list that PSF will use. It also improves performance since you can just fill in the libraries that need to be searched and leave out the ones that do not need to be searched.

Part 4. Other printing functions available on OS/400

Chapter 11. Working with ASCII Lexlink protocol LAN-attached printers

This section of the manual discusses ASCII LAN-attached printers that use the Lexlink protocol

Information provided is:
- Benefits of using ASCII LAN-attached printers
- How ASCII LAN-attached printers are supported on the iSeries server
- Restrictions when using ASCII LAN-attached printers
- Configuration parameters that support ASCII LAN-attached printers
- Configuring and starting ASCII LAN-attached printers

Current support requires that ASCII printers be attached to the IBM 4033 LAN adapter device or MarkNet XLe device, or the printer have a MarkNet or MarkNet XL Internal Network Adapter (INA) card resident in the printer. (The IBM 4039 Printer is an example of a printer with an INA card.)

OS/400 provides all the support for ASCII printers attached to the LAN. Once the physical environment is created, the ASCII LAN-attached printers operate similar to any other ASCII printer attached to an iSeries server.

Benefits of using ASCII LAN-attached printers

ASCII LAN-attached printers offer many benefits over printers attached using other methods. Some of these benefits are:
- Low-cost

  Typically, ASCII printers are less expensive than larger SCS or IPDS printer. With new technology, ASCII printers are fast becoming capable of performing advanced functions formerly found only on SCS or IPDS printers.
- Central printing and portability

  Using a Local Area Network (LAN) to attach ASCII printers allows you to position the printer at the point where most of your printing requirements are. If this requirement changes, the printer can be physically moved to another point on the LAN.
- Sharing the printer between different operating systems

  Sharing, in this case, actually has two benefits. The first is that any user attached to the LAN can send print jobs to the printer. The second is a LAN-attached ASCII printer can perform printing for an iSeries server, an RS/6000*, or a PS/2 running OS/2.

  Note: The RS/6000 and PS/2 must have the Network Print Adapter Utility installed. The utility is offered by Lexmark and IBM. The utility allows PCs and RS/6000s to use the 4033 LAN adapter device to attach to the LAN. An IBM 4039 printer with the INA card can also attach to the LAN
- Eliminating emulation requirements

  Before the ASCII LAN-attached printer function became available, ASCII printers used with iSeries servers were predominantly used through an emulation program and attached to a physical device such as a PC.

ASCII printers attached to the LAN can receive print jobs composed of the SCS, USERASCII, or AFP data streams. The SCS or AFP data stream is sent through the host print transform function. The host print transform function transforms an SCS or AFP data stream to an ASCII data stream. The USERASCII data stream is sent directly to the printer.

In general, you can access LAN printers attached to the devices through bridges instead of routers. For example the IBM 4033, the MarkNet XLe, the MarkNet and the MarkNet XL support the IBM 8209 LAN bridge.

How ASCII LAN-attached printing works

Figure 2 provides a physical view of how ASCII LAN-attached printers connect to the iSeries server.

RV3H006-0
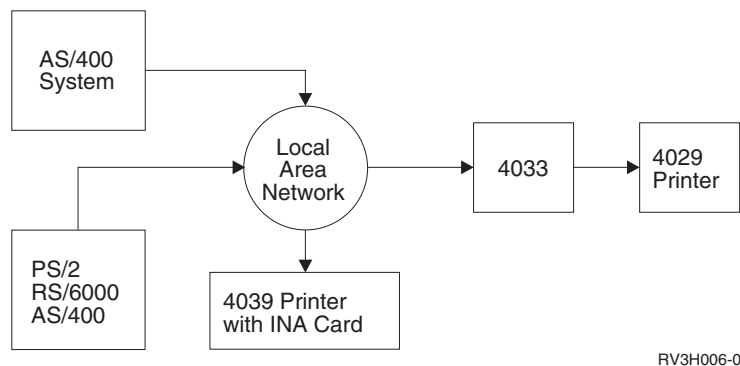
Figure 2. Physical view of ASCII LAN-attached printers

Figure 3 on page 253 provides a software view of how ASCII LAN-attached printers appear to the iSeries server.

```
        ┌─────────────┐
        │ Run         │
        │ STRPRTWTR   │
        │ Command     │
        └─────────────┘
```

User Process
- -
Writer Process

```
        ┌─────────────────┐      ┌──────────────────┐
        │                 │◄─────│ Printer Device   │
        │ Printer Writer  │      │ Description       │
        │ Job             │      │ -SWTLINLST        │
        └─────────────────┘      └──────────────────┘
```

```
┌──────────────────────────────────────────────────────────────────┐
│ Communications Line    Network Controller    Network Device        │
│  ┌──────────────┐       ┌──────────────┐      ┌──────────────┐     │
│  │ Line         │       │              │      │              │     │
│  │ Description  │       │ Controller   │      │ Device       │     │
│  │  -Token Ring │       │ Description  │      │ Description   │     │
│  │  -Ethernet   │       │              │      │              │     │
│  └──────────────┘       └──────────────┘      └──────────────┘     │
│                                                                    │
│  Configuration descriptions that provide connection to the LAN     │
└──────────────────────────────────────────────────────────────────┘
```

```
        ┌───────────┐              ┌──────────────┐
        │  Local    │              │              │
        │  Area     │──────────────│     4033     │
        │  Network  │              │              │
        │  (LAN)    │              └──────────────┘
        └───────────┘
```

```
  ┌──────────────────┐          ┌──────────────┐
  │ ASCII Printer    │          │ IBM          │
  │ with INA Card    │          │ 4029 Printer │
  │ (IBM 4039)       │          │              │
  └──────────────────┘          └──────────────┘
```

RV3H051-0
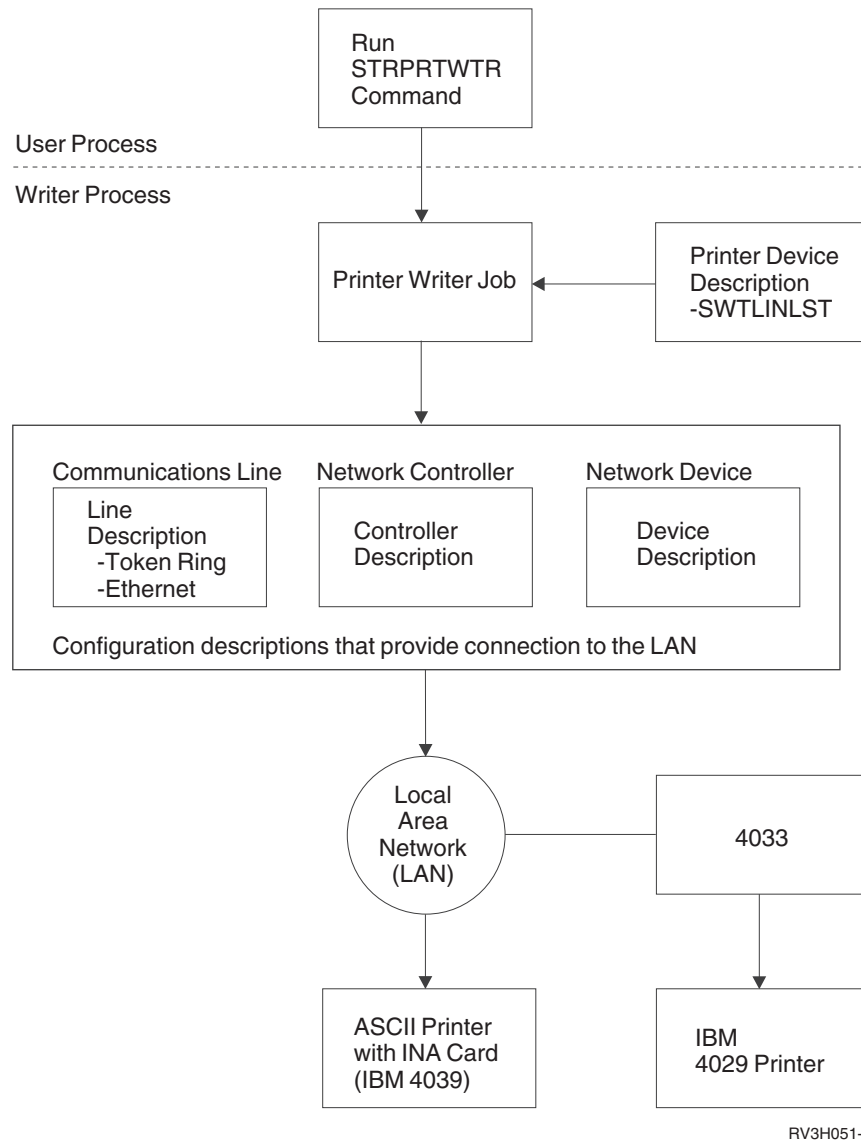
Figure 3. Software view of ASCII LAN-attached printers

Once the physical environment is in place and the correct configuration for the ASCII printer complete, printing operations can begin. As with other printers, the Start Printer Writer (STRPRTWTR) command is used to begin printing operations.

Note: The printer device description must be varied on before the writer can be started.

When the STRPRTWTR command is run, the printer's device description switched line list (SWTLINLST) parameter is used to identify available communications lines. These communications lines must be either Ethernet or Token Ring. The line descriptions must already exist and they must be varied on to be eligible for LAN printing. The source service access point (SSAP) parameter value of the line description must contain an SSAP value of 12 and the SSAP type value must be *NONSNA.

Notes:

1. On existing communications lines, the SSAP value of 12 and the SSAP Type value of *NONSNA must be added to the SSAP list before using ASCII-LAN attached printers.
2. If the printer writer ends with message CPI400C and a reason code of 1, the line being used does not have SSAP 12 configured.

The line descriptions provide a network controller (NETCTL) parameter. If this parameter has a controller specified, and if the controller is available, it is attached to the line description.

In turn, the controller description has a device (DEV) parameter. If the DEV parameter has a device specified, and if the device is available, it is attached to the controller description.

If the network controller and network device are not available, they are automatically created when the STRPRTWTR command is run. The network controller and network device description objects are not deleted when the writer session is ended. Each time the STRPRTWTR command is run, and the network controller is not available, another network controller and network device is created.

Note: The LAN adapter device is locked when the first spooled file begins to print. The inactivity timer parameter in the printer device description is used to determine when to release the adapter. Releasing the adapter enables other users on the LAN to use the printer.

Due to a hardware restriction on the 4033 LAN adapter device, it is suggested that *ATTACH or *NOMAX be used as the value for the inactivity parameter. Thus, on a 4033, the lock on the adapter device is not given up shortly after a spooled file prints but rather when the writer is ended.

Restrictions when using ASCII LAN-attached printers

The following list contains restrictions for ASCII LAN-attached printers:

- All ASCII printers must be attached to the IBM 4033 LAN adapter device or the MarkNet XLe device, or the printer must have a MarkNet or MarkNet XL Internal Network Adapter (INA) card.
- Only spooled files with device type attributes of *SCS, *USERASCII, or *AFPDS are supported.
- Direct printing (SPOOL parameter value = *NO on the CRTPRTF, CHGPRTF, and OVRPRTF commands) is not allowed to ASCII LAN-attached printers.
- Printer sharing through the allow direct print (ALWDRTPRT) parameter on the STRPRTWTR command is not supported. Printer sharing means having the SPOOL parameter value = *NO on the CRTPRTF command and the ALWDRTPRT parameter value = *YES on the STRPRTWTR command.
- Only Token Ring or Ethernet communications lines (values in the Switched Line List (SWTLINLST) parameter of the printer device description) are supported.
- Error recovery will not be detailed.
- Spooled files of type other than *SCS, *USERASCII, or *AFPDS that reach an output queue associated with a ASCII LAN-attached printer are held. A message indicating the spooled file is held is sent to the message queue specified on the STRPRTWTR command.

Line, controller, and device description parameters that support ASCII LAN-attached printers

To correctly configure descriptions (line, network controller, network device, and printer device) certain parameters on these descriptions must have specific values.

Line description parameters that support ASCII LAN-attached printers

All lines must be either Token Ring or Ethernet.

Note: The Ethernet standard parameter value must be IEEE8023 or *ALL.

Not all of the parameters that make up the line description are discussed here. Only the parameters required to support ASCII LAN-attached printers are shown.

SSAP

Source service access point

For the IBM 4033, MarkNet and MarkNet XL, specify *SYSGEN, or 12. For the MarkNet XLe, specify 12, 16 and 1A.

Note: If the required SSAP values are not added to existing communications lines used for ASCII LAN-attached printers, message CPI400C is sent to the message queue with a reason code of 1.

SSAP type

Source service access point (TYPE)

This value must be *NONSNA. This means SNA communications cannot be used.

Note: If the line description already exists you can use the Change Line Description (Token-Ring) (CHGLINTRN) or Change Line Description (Ethernet) (CHGLINETH) commands to update the line descriptions with the required SSAP values.

NETCTL

Network controller

The name of the network controller this line attaches to.

Controller and device description parameters that support ASCII LAN-attached printers

Network controller and network device descriptions are needed to complete the communications connection to the ASCII LAN-attached printer.

If they don't exist, they are created automatically when the STRPRTWTR command is run for a ASCII printer attached to the LAN.

Network controllers must be used. They must exist or you can create them using the Create Controller Network (CRTCTLNET) command.

Not all of the parameters that make up the controller description are discussed here. Only the parameters required to support ASCII LAN-attached printers are shown.

DEV
> Device name

> The name of the device this controller attaches to.

Network device description parameters that support ASCII LAN-attached printers

A network device must be used. The Create Device Network (CRTDEVNET) command must be used. This command creates a network device necessary to connect to the LAN. Only the parameters required to support ASCII LAN-attached printers are shown.

TYPE
> Device type

> This device is attached to a network that is supported by a program using the user-defined communications Application Program Interface (APIs).

Printer device description parameters that support ASCII LAN-attached printers

An ASCII printer device must be configured. The Create Device Description (Printer) (CRTDEVPRT) command must be used.

ASCII LAN-attached printers are enabled by selecting certain values for parameters in the printer device description. Not all of the parameters that make up the printer device description are discussed here. Only the parameters required to support ASCII LAN-attached printers are shown.

The following parameters, on the printer device description, are used to configure ASCII LAN-attached printers.

DEVCLS
> Device class

> Specifies the device class for the printer. This value must be *LAN and it indicates the ASCII printer is connected to a local area network (LAN).

TYPE
> Device type

> Specifies the type of printer this device configuration represents. For example, you can specify 3812 as IBM 3812 emulation is used.

MODEL
> Device model.

> Specifies the model of the device. For example, you can specify 1 as IBM 3812 Model 1 emulation is used.

SWTLINLST
> Switched line list

> Specifies the name of the switched communications lines to which the printer is associated when DEVCLS(*LAN) is specified. A maximum of 8 switched communications lines can be specified.

ADPTADR
> LAN remote adapter address (printer address)

Specifies the 12-character hexadecimal LAN address of the ASCII printer when DEVCLS(*LAN) is specified. Valid values range from 000000000001 through FFFFFFFFFFFE. This address is available with the LAN adapter.

Note: The adapter address for the 4033 LAN-adapter is printed on the box it is packaged in.

The address for a 4039 printer with an INA card can be found using the printer's operator panel. The address for the MarkNet XLe is printed on the back side of the device.

It is possible for the adapter address to be changed using the Network Print Utility. If the adapter address is changed, the new address must be the value specified for the ADPTADR parameter in the printer's device description.

LANATTACH
LAN attachment.

Specifies the attachment of the printer when DEVCLS(*LAN) is specified. This value must be *LEXLINK.

ADPTTYPE
Adapter type

Specifies the type of LAN printer adapter to be used when DEVCLS(*LAN) is specified.

*INTERNAL
The printer has an internal LAN adapter. For example, a printer with an INA card installed.

*EXTERNAL
The printer has an external LAN adapter. For example, a printer is connected to the IBM 4033 LAN adapter or the MarkNet XLe

ADPTCNNTYP
Adapter connection type

Specifies the type of ports supported by the external LAN printer adapter when DEVCLS(*LAN) and ADPTTYPE(*EXTERNAL) are specified.

*PARALLEL
The printer is attached to the adapter using the parallel port.

*SERIAL
The serial port on the adapter is used to communicate with the attached printer.

ATTACH
Physical attachment

For ASCII LAN-attached printers, specifies the physical attachment of the LAN adapter.

Note: This parameter is used only if ADPTTYPE is *EXTERNAL and ADPTCNNTYPE is *SERIAL.

For ASCII LAN-attached printers, possible values are:

*DIRECT
Specifies EIA-232 direct attachment.

*WIRE4
> Specifies EIA-232 4-wire attachment.

PORT
> Port

> For ASCII LAN-attached devices, if there is more than one serial or parallel port available, this parameter specifies which port is used. Possible values are 0 through 17.

> For the MarkNet XLe the following values are used:

> **Value**   **Port**

> **0**   Serial

> **1**   Parallel 1

> **2**   Parallel 2

> **Note:** This parameter is not used if the printer is attached to a 4033 LAN adapter.

INACTTMR
> Inactivity timer

> Specifies an inactivity timer (time-out) value. For ASCII LAN-attached printers, this value indicates the amount of time the printer writer keeps a lock on the device before releasing it.

> **Note:** This parameter is not used if the printer is attached to a 4033 LAN adapter.

> Possible values are:
> - *ATTACH

>   If the printer has an INA card, 15 seconds must pass before releasing the adapter back to the network.

>   If the Adapter type is specified as *EXTERNAL, *ATTACH sets *NOMAX as the value for releasing the adapter back to the network. *NOMAX is used if the printer is attached to a 4033 because the 4033 does not support sharing between jobs or systems. Since the MarkNet XLe supports sharing, it is recommended that you set the INACTTMR value to *SEC15.
> - *NOMAX - Maximum inactivity time is not tracked.
> - *SEC15 - A 15-second time-out period is used.
> - *SEC30 - A 30-second time-out period is used.
> - Inactivity timer - The range to select from is 1 through 30 minutes.

ACTTMR
> Activation timer

> Specifies the amount of time (in seconds) to wait for the printer to respond to the activation request from the host system. If the printer does not respond within this time, it is considered not available and a cancel/retry message is issued to the user.

> For example, if the activation timer value is 120 seconds, the writer attempts to lock the adapter every 15 seconds for 120 seconds. After 120 seconds elapses, a cancel/retry message is issued. A cancel reply to the message cancels the writer. A retry reply causes the writer to attempt to lock the adapter again every 15 seconds for 120 seconds. If the lock attempt fails the cancel/reply

message is issued again. If the message is not answered in 5 minutes, the system automatically attempts to lock the adapter. There is also an allocate (A) reply which releases the 4033 from any system. Before selecting the A reply for a printer with an INA card, the printer must be powered off and back on.

Possible values are:

170   The printer waits 170 seconds. This is the default.

Activation -timer
Specify a number indicating the number of seconds before the device is considered not available. When this number of seconds has elapsed, an inquiry message is issued.

If DEVCLS(*LAN) is specified, valid values range from 1 through 2550.

TRANSFORM
Host print transform function

The host print transform function value is defaulted to *YES if the device class (DEVCLS) parameter value is *LAN and LANATTACH(*LEXLINK).

*NO   The printer does not use the host print transform function.

*YES   The printer uses the host print transform function.

USRDTATFM
User data transform.

Specifies the qualified name of a user data transform program that is used to transform the spooled file data. The possible values are:

*NONE
No user-defined data transform program name is specified.

name   Specifies the name of the user data transform program. It can be qualified by one of the following library values:

*LIBL   All libraries in the job's library list are searched until the first match is found.

*CURLIB
The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

Library-name
Specifies the library to be searched.

Configuring and starting ASCII LAN-attached printers

This example shows how to configure and then start an IBM 4039 printer with an INA card resident in the printer. Use the following list to identify parameters and values for configuring the device description. Use default values for the other parameters.

- Device description - PRT4039
- Device class - *LAN
- Device type - 3812
- Device model - 1
- Switched line list

The line descriptions don't need to exist to create the device description. However, they must be entered as values in the switched line list parameter.

- LAN remote adapter address - comes with the LAN adapter

  Note: The adapter address for the 4033 LAN-adapter is printed on the box it is packaged in.

  The address for a 4039 printer with an INA card can be found using the printer's operator panel.

  It is possible for the adapter address to be changed using the Network Print Utility. If the adapter address is changed, the new address must be the value specified for the ADPTADR parameter in the printer's device description.

- Adapter type - *INTERNAL (the 4039 printer has an INA card)
- Font - a font must be specified
- Manufacturer type and model - *IBM4039HP
- Text 'description' - Device description for an ASCII LAN-attached printer

Note: If *SERIAL is selected as the adapter connection type the following parameters need to be used:
  - Line speed
  - Word length
  - Parity
  - Stop bits
  - Physical attachment

Type CRTDEVPRT and prompt (F4). The following screen appears.

```
                  Create Device Desc (Printer) (CRTDEVPRT)

 Type choices, press Enter.

 Device description . . . . . . . > PRT4039        Name
 Device class . . . . . . . . . . > *LAN           *LCL, *RMT, *VRT, *SNPT, *LAN
 Device type  . . . . . . . . . . > 3812           3287, 3812, 4019, 4201...
 Device model . . . . . . . . . . > 1              0, 1, 2, 3, 4, 10, 13, 200...
 Switched line list . . . . . . . > TRNLINE    Name
               + for more values
 LAN remote adapter address . . . > 8FFFFFFFFFFF  000000000001-FFFFFFFFFFFE
 Adapter type . . . . . . . . . . > *INTERNAL      *INTERNAL, *EXTERNAL
 Emulated twinaxial device  . . . > 3812           3812, 5219, 5224, 5256
 Font:
   Identifier . . . . . . . . . .                  3, 5, 11, 12, 13, 18, 19...
   Point size . . . . . . . . . .       *NONE      000.1-999.9, *NONE
 Manufacturer type and model  . . > *IBM4039HP
 Text 'description' - Device description for an ASCII LAN-attached


                                                                    More...
 F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
 F13=How to use this display        F24=More keys
```

To start PRT4039 type STRPRTWTR and prompt (F4). The following screen appears. Type PRT4039 for the Printer parameter value and type the name of the output queue from which you want to print spooled output files.

Spooled files from MYOUTQ will print on the printer named PRT4039.

Chapter 12. Working with ASCII TCP/IP network-attached printers

This chapter provides the following information:
- Benefits of using ASCII TCP/IP network-attached printers
- Support for ASCII TCP/IP network-attached printers on iSeries servers.
- Restrictions when using ASCII TCP/IP network-attached printers
- Configuration parameters that support ASCII TCP/IP network-attached printers
- Configuring and starting ASCII TCP/IP network-attached printers

For network printing, the iSeries server supports HP Printer Job Language (PJL), Simple Network Management Protocol (SNMP), and Internet Printing Protocol (IPP).

HP printer job language (PJL)

To print using PJL, your ASCII printer must be attached to the TCP/IP network and accept print data on a TCP/IP port. The printer must support PCL5e and the network interface (either internal or external) must support bidirectional PJL on a TCP/IP port. OS/400 uses the PJL commands to obtain printer status.

OS/400 provides all the support for ASCII printers that are attached to the TCP/IP network. Once the physical environment is created, the ASCII TCP/IP network-attached printers operate similarly to any other ASCII printer that is attached to an iSeries server.

Simple network management protocol (SNMP)

To print using SNMP, your ASCII printer must be attachd to the TCP/IP network and accept print data on a TCP/IP port. The SNMP management API communicates using a different TCP/IP port than the print data stream. OS/400 uses the SNMP management API to obtain printer status. You can specify the SNMP print driver by using the system driver program parameter of the printer device description. Once the physical environment is created, your ASCII TCP/IP network-attached printer operates similarly to any other iSeries server-attached ASCII printer.

Internet Printing Protocol (IPP)

To print using IPP, your ASCII printer must be attached to the TCP/IP network and accept print data on the well known IPP port, port 631. You can specify the IPP print driver by using the system driver program parameter of the printer device description. Once the physical environment is created, your ASCII TCP/IP network-attached printer operates similarly to any other iSeries server-attached ASCII printer.

Benefits of using ASCII TCP/IP network-attached printers

ASCII TCP/IP network-attached printers offer many benefits over printers that are attached with other methods. Some of these benefits are:
- Low-cost

Typically, ASCII printers are less expensive printers. The low cost and new
function provide increased printer flexibility.

- Central printing and portability

  Attaching ASCII printers to a TCP/IP network allows you to position the printer
  at the point where most of your printing requirements are. If this requirement
  changes, you can physically move the printer to another point on the TCP/IP
  network.

- Sharing the printer between different operating systems

  Any user that is attached to the network can send print jobs to the printer. A
  TCP/IP network-attached ASCII printer can perform printing for an iSeries
  server concurrently with printing for other IBM and non-IBM operating systems.

- Eliminating emulation requirements

  Before the ASCII TCP/IP network-attached printer function became available,
  OS/400 users often had to use the ASCII printers through an emulation program
  and a physical device such as a PC.

  ASCII printers that are attached to the network can receive print jobs that
  contain the SCS, USERASCII, or AFP data streams. You can send the SCS or AFP
  data stream through the host print transform function. The host print transform
  function transforms an SCS or AFP data stream to an ASCII data stream. The
  USERASCII data stream is sent directly to the printer.

How ASCII TCP/IP network-attached printing works

Figure 4 provides a physical view of how ASCII TCP/IP network-attached printers
connect to the iSeries server.
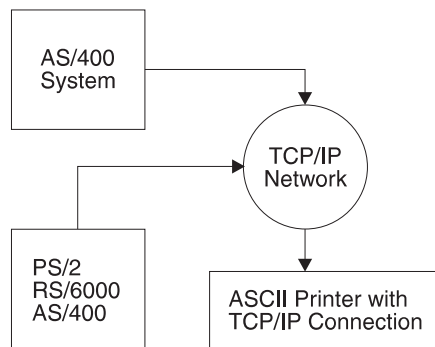
Figure 4. Physical view of ASCII TCP/IP network-attached printers

Figure 5 on page 263 and 3 provide software views of how ASCII TCP/IP
network-attached printers appear to the iSeries server.
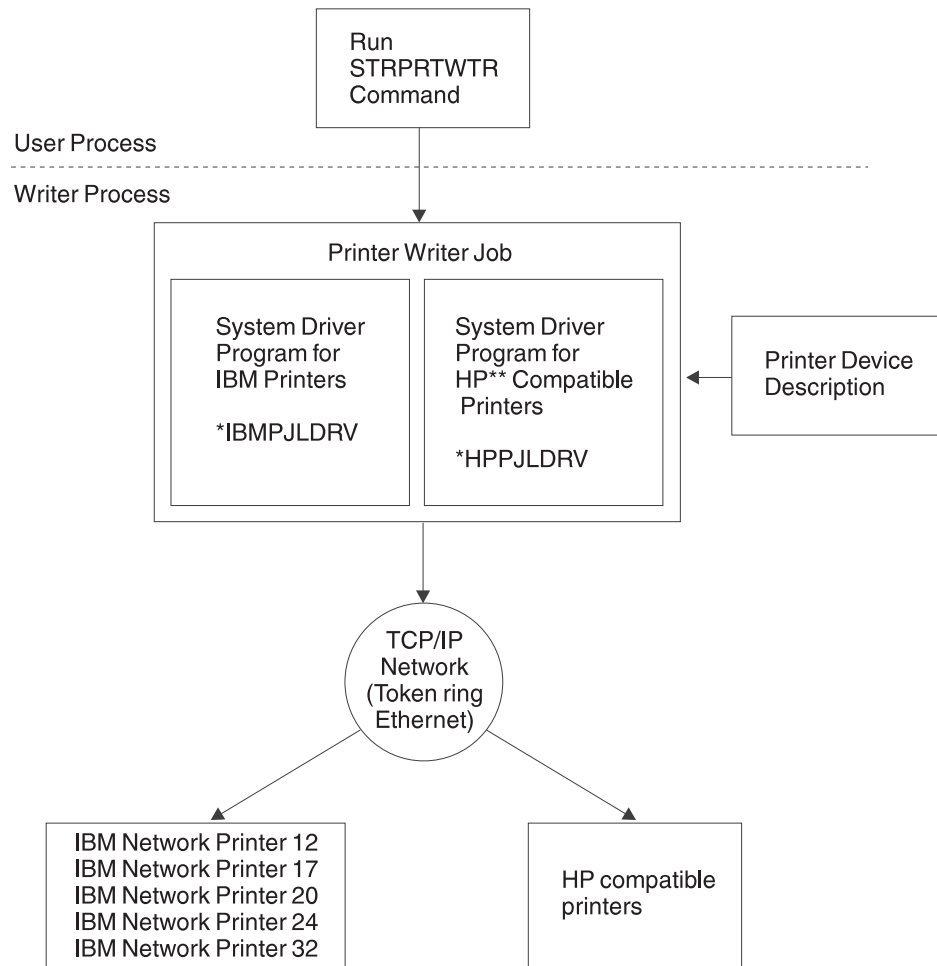
Figure 5. Software view of ASCII TCP/IP network-attached printers using PJL

```
                    ┌──────────────────┐
                    │   Run            │
                    │   STRPRTWTR      │
                    │   command        │
                    └──────────────────┘
User process                   │
- - - - - - - - - - - - - - - -│- - - - - - - - - - - - - - - - - - - - - - -
Writer process                 │
                               ▼
          ┌─────────────────────────────────────┐
          │   Printer writer job                 │
          │   ┌───────────────────┐              │
          │   │                   │   ┌──────────────────┐
          │   │ System driver     │   │                  │
          │   │ program           │◄──│ Printer device   │
          │   │                   │   │ description      │
          │   │ *IBMSNMPDRV       │   └──────────────────┘
          │   │                   │              │
          │   └───────────────────┘              │
          └─────────────────────────────────────┘
                               │
                               ▼
                          ╱─────────╲
                         │  TCP/IP   │
                         │  Network  │
                         │(Token ring│
                         │ Ethernet) │
                          ╲─────────╱
                               │
                               ▼
          ┌─────────────────────────────────────┐
          │   Supported printers*                │
          └─────────────────────────────────────┘
                                        RBAFT501-1
```
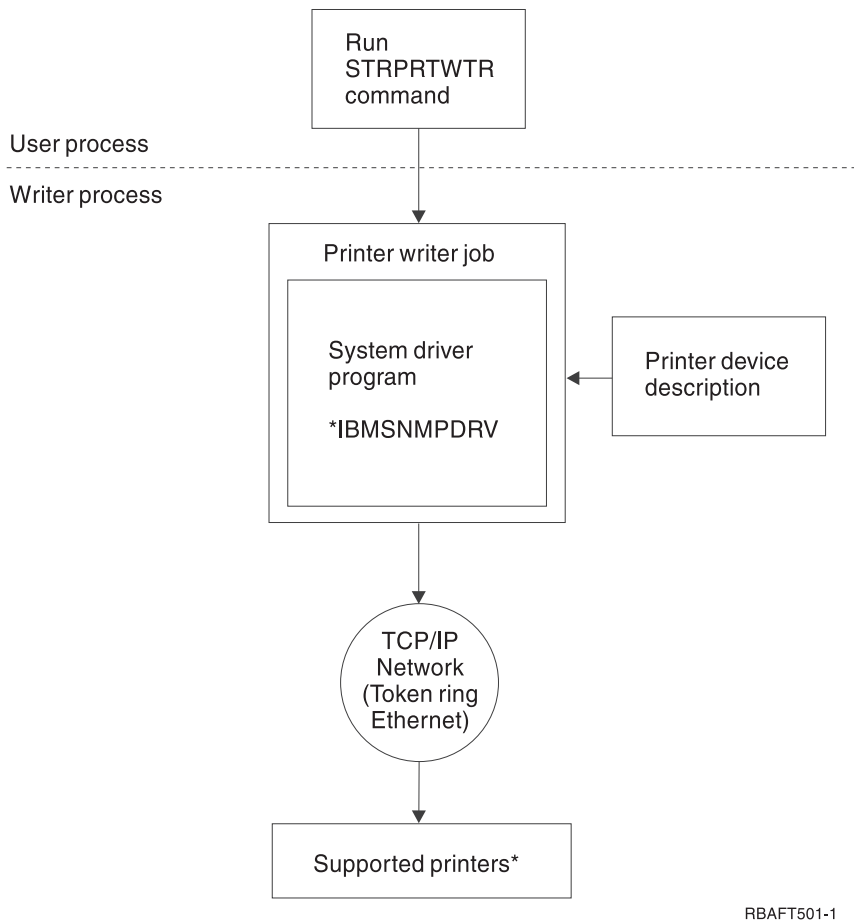
Note: See your printer manufacturer's manual to determine if your printer is capable of SNMP printing. Also, see "Common restrictions" and "SNMP restrictions" on page 265 for additional requirements for SNMP printing.

Figure 6. Software view of ASCII TCP/IP network-attached printers using SNMP

Once the OS/400 TCP/IP configuration for the ASCII printer is complete and the physical environment is in place, the printing operations can begin. As with other printers, you can use Start Printer Writer (STRPRTWTR) command to start printing operations.

Note: You must vary on the printer device description before starting the printer writer.

When you start the STRPRTWTR command, the OS/400 establishes communications with the printer by opening a TCP/IP connection with the printer. The port number of the printer is in the printer device description.

Restrictions when using ASCII TCP/IP network-attached printers

Common restrictions
These restrictions apply to all ASCII TCP/IP network-attached printers:
- They must be able to attach directly to a TCP/IP network and accept print data on a TCP/IP port number.

- They only support spooled files with device type attributes of *SCS, *USERASCII, or *AFPDS.
- They do not allow direct printing (SPOOL parameter value = *NO on the CRTPRTF, CHGPRTF, and OVRPRTF commands).
- They do not support sharing printers through the direct print (ALWDRTPRT) parameter on the STRPRTWTR command. Printer sharing means specifying the SPOOL parameter value = *NO on the CRTPRTF command and the ALWDRTPRT parameter value = *YES on the STRPRTWTR command.
- They do not provide detailed error recovery message.
- If an output queue is associated with an ASCII TCP/IP network-attached printer, it will hold any spooled file type except *SCS, *USERASCII, or *AFPDS when *TRANSFORM(*YES) has been specified on the Printer Device Description. When a spooled file is held, a spooled file status message will be sent to the message queue that is specified on the STRPRTWTR command.

PJL restrictions

In addition to the common restrictions listed above, the following restrictions apply only to ASCII TCP/IP network-attached printers that use PJL:

- They must be able to accept HP PJL commands on the same TCP/IP port on which the print data is received .

Additional PJL requirements are listed in "HP printer job language (PJL)" on page 261 section.

SNMP restrictions

In addition to the common restrictions listed above, the following restrictions apply only to ASCII TCP/IP network-attached printers that use SNMP :

- They must support the industry standard Host Resource Management Information Base (RFC 1514). It is highly recommended, but not required, that these printers support the Printer Management Information Base (RFC 1759).
- If the printer is connected to a network adapter, the adapter must be compatible with the printer. The adapter must also support the Host Resource Management Information Base. Consult the manufacturer of the adapter to determine if it supports the printer. Some network adapters do not support all printers.
- If the printer is attached to an external network adapter that has more than one port, the printer should be connected to the first parallel port. There should be no other SNMP capable devices attached to the adapter.
- You must set the printer, and any adapter that is connected to it, to a community name of public. Public is usually the default community setting from the factory. Community names are a means of access control for SNMP devices. Read-only access to the public community is sufficient.

IPP restrictions

In addition to the common restrictions listed above, the following restrictions apply only to ASCII TCP/IP network-attached printers that use IPP:

- They must support the industry standard Internet Printing Protocol (IPP) as defined in the following:
  - RFC 2911 IPP/1.1: Model and Semantics
  - RFC 2910 IPP/1.1: Encoding and Transport
- To configure for secure connections, the printers or print servers must support TLS or SSL.

Printer device description parameters that support ASCII TCP/IP network-attached printers

You must use the Create Device Description (Printer) (CRTDEVPRT) command to configure an ASCII printer device.

You can enable ASCII TCP/IP network-attached printers by selecting certain values for parameters in the printer device description. Not all of the parameters that make up the printer device description are discussed here. Only the parameters required to support ASCII TCP/IP network-attached printers are shown.

You can use the following parameters to configure ASCII TCP/IP network-attached printers.

DEVCLS
   Device class

   It specifies the device class for the printer. This value must be *LAN. It indicates that the ASCII printer is connected to network of any type.

TYPE
   Device type

   It specifies the type of printer this device configuration represents. For example, you can specify 3812 as IBM 3812 emulation is used.

MODEL
   Device model.

   It specifies the model of the device. For example, you can specify 1 as IBM 3812 Model 1 emulation is used.

LANATTACH
   LAN attachment.

   You need to specify *IP for this parameter.

PORT
   Port

   Many printers accept print data on port 9100. For IBM network printers, use port 2501. Refer to the printer manual, or contact the manufacturer of your printer to get the port information.

   For printers supporting IPP, use port 631.

INACTTMR
   Inactivity timer

   It specifies an inactivity timer (time-out) value. This value indicates the amount of time the printer writer keeps a TCP/IP connection open to the printer before closing it, when there are no files in a ready status. Possible values are:

   *ATTACH
      A 15-second time-out period is used.

   *NOMAX
      Maximum inactivity time is not tracked. The printer writer keeps the connection to the printer open until the writer is ended.

   *SEC15
      A 15-second time-out period is used.

*SEC30
> A 30-second time-out period is used.

Inactivity timer
> The range to select from is 1 through 30 seconds.

ACTTMR
Activation timer

It specifies the amount of time (in seconds) to wait for the printer to respond to the connection request from the host system. It is also used to timeout when waiting for a response from the printer after sending data or PJL commands. If the printer does not respond within this time, it is considered not available and a cancel/retry message appears. The user must end and restart the writer when this happens. For example, if the activation timer value is 120 seconds, the printer writer attempts to establish a connection to the printer and waits for 120 seconds. After 120 seconds elapses, the cancel/retry message appears. A cancel reply to the message cancels the writer. A retry reply causes the writer to attempt to establish a connection again and waits for 120 seconds. If the attempt fails, the message appears again. The speed of the network influences what the setting should be.

The possible values are:

170   The printer waits for 170 seconds. This is the default value.

Activation timer
> It specifies the number of seconds before the device is not available. When the number of seconds has elapsed, an inquiry message appears. If DEVCLS(*LAN) is specified, valid values can range from 1 through 2550.

TRANSFORM
Host print transform function

The value of the host print transform function defaults to *YES if the device class(DEVCLS) parameter value is *LAN and LANATTACH(*IP).

*NO   The printer does not use the host print transform function.

*YES   The printer uses the host print transform function.

RMTLOCNAME
Remote location name

It specifies either the TCP/IP address of the printer or the name of the printer if a name is configured in the TCP/IP host table.

SYSDRVPGM
System driver program.

It specifies the print driver type for this configuration.

For IBM printers that use PJL, this value should be *IBMPJLDRV. For other TCP/IP-attached printers that use PJL, this value should be *HPPJLDRV.

For TCP/IP-attached printers that use SNMP, this value should be *IBMSNMPDRV.

For TCP/IP-attached printers that use IPP, this value should be *IBMIPPDRV.

USRDFNOPT
User-defined options.

It specifies, for spooled output only, one or more user-defined options to user applications or user-specified programs or by user-specified programs that process spooled files. You can specify a maximum of four user-defined options. The possible values are:

*NONE
>   No user-defined option is specified.

user-defined option
>   If you specify *IBMSNMPDRV for the SYSDRVPGM parameter, you can specify *IBMSHRCNN for the user-defined option. The printer writer will open and close the connection to the data port on the printer for every copy of every spooled file. This allows multiple printer writers and systems to access the printer even when there are additional copies or files in a ready status to be processed. If you specify *IBMSHRCNN, the INACTTMR parameter is ignored.
>
>   **Note:** You must specify *IBMSHRCNN for the IBM Infoprint 21 printer.

USRDTATFM
>   User data transform.
>
>   It specifies the qualified name of a user data transform program that transforms the spooled file data. The possible values are:
>
>   ***NONE**
>   >   No user-defined data transform program name is specified.
>
>   **name** It specifies the name of the user data transform program. One of the following library values can qualify it:
>
>   >   ***LIBL** All libraries in the job's library list are searched until the first match is found.
>   >
>   >   ***CURLIB**
>   >   >   The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.
>   >
>   >   **Library-name**
>   >   >   It specifies the library to be searched.

SECURECNN
>   Secure connection
>
>   This parameter is used only by the IPP driver. It specifies whether a secure connection is established with the printer. A secure connection provides an encrypted communications session to ensure print data that passes over the connection remains private.
>
>   ***NO**   Connection with printer is not secure.
>   ***YES**   Connection with printer is secure. The printer must support SSL (Secure Sockets Layer) or TLS (Transport Layer Security) and must have a system digital certificate.

VLDL
>   Validation list
>
>   This parameter is used only by the IPP driver. It specifies a validation list that is used if the printer requests authentication. See "Setting up validation lists for the IPP print driver" on page 269 for details.

Setting up validation lists for the IPP print driver

An IPP printer or IPP print server may require user authentication on requests from the iSeries IPP print driver. If this is the case, the validation list, specified in the VLDL parameter of the printer device description, is used to look up the password for a given user ID.

A validation list holds pairs of user IDs and passwords. The IPP print driver checks the validation list for an entry in the following order:

1. The iSeries user profile that owns the spooled file
2. The iSeries printer device description name
3. The iSeries system name

If a match is found, the password will be retrieved for that entry and passed in subsequent requests to the printer. You might choose not to include individual user IDs in the list and instead provide the printer device description name or the system name as a generic entry to be used by multiple users.

If a match is not found, then the user authentication cannot be done.

Use the Create Validation List (CRTVLDL) command to create an empty validation list, as shown in the following example:

```
CRTVLDL VLDL(MYLIB/MYUSRS) AUT(*EXCLUDE) TEXT('My users')
```

This command creates an empty validation list named MYUSRS in the MYLIB library. The validation list is specified in the VLDL parameter of the printer device description. See the CL category in the iSeries Information Center for details. Use the AUT parameter to limit user access to the validation list.

After you create the validation list to be used by the IPP print driver, you populate the list with entries consisting of a user ID and a password that the system encrypts when the list is stored. You can add, change, and remove entries with the Validation List APIs. See the APIs category in the iSeries Information Center for details.

Use the QsyAddValidationLstEntry() API to add an entry to a validation list. You must specify the QsyEncryptData attribute when adding entries in order to permit the retrieval of the encrytped password.

Configuring and starting ASCII TCP/IP network-attached printers

This example shows how to configure and then start an IBM 4312 printer. The network interface card must support the TCP/IP protocol and be active. You can use the following list to identify parameters and values for configuring the device description. For the other parameters, use the default values.

- Device description - PRT4312
- Device class - *LAN
- Device type - 3812
- Device model - 1
- LAN attachment type - *IP
- Port number - 2501
- Font - a font must be specified
- Manufacturer type and model - *IBM4312

- Text 'description' - Device description for an ASCII network-attached printer
- Remote location - 9.5.11.334
- System driver program - Specify *IBMPJLDRV if you are using PJL. Specify *IBMSNMPDRV if you are using SNMP.

To configure an IBM 4312 printer, type CRTDEVPRT and press F4=Prompt. The following display appears.

```
                Create Device Desc (Printer) (CRTDEVPRT)

Type choices, press Enter.

Device description . . . . . . . > PRT4312    Name
Device class . . . . . . . . . . > *LAN       *LCL, *RMT, *VRT, *SNPT, *LAN
Device type  . . . . . . . . . . > 3812       3287, 3812, 4019, 4201...
Device model . . . . . . . . . . > 1          0, 1, 2, 3, 4, 10, 13, 301...
LAN attachment . . . . . . . . . > *IP        *LEXLINK, *IP, *USRDFN
Port number  . . . . . . . . . . > 2501       0-65535
Online at IPL . . . . . . . . . . *YES          *YES, *NO
Font:
  Identifier . . . . . . . . . . > 11         3, 5, 11, 12, 13, 18, 19...
  Point size . . . . . . . . . .  *NONE       000.1-999.9, *NONE
Form feed  . . . . . . . . . . .  *Type       *Type, *CONT, *CUT, *AUTOCUT
Separator drawer . . . . . . . .  *FILE       *1-225, *FILE
Separator program  . . . . . . .  *NONE       Name *NONE
  Library  . . . . . . . . . . .  *LIBL       Name *LIBL,*CURLIB
Printer error message  . . . . .  *INQ        *INQ, *INFO

                                                            More...
F3=Exit   F4=Prompt   F5=Refresh   F10=Additional parameters   F12=Cancel
F13=How to use this display       F24=More keys
```

To start an IBM 4312 printer, type STRPRTWTR and press F4=Prompt. The following display appears. Type PRT4312 for the Printer parameter value and type the name of the output queue from which you want to print spooled output files.

Spooled files from MYOUTQ will print on printer PRT4312.

Chapter 13. Working with the host print transform function

This chapter provides information about the host print transform function and how to enable it to work with existing emulators that support ASCII printers attached to the iSeries server. The host print transform function is an OS/400 function that converts an SNA character string (SCS) or Advanced function print (AFP) data stream into an ASCII data stream. The ASCII data stream is then formatted and sent to an ASCII printer through one or more hardware connections, such as iSeries Access for Windows, or the 3477 or 3487 work stations. This single location of the conversion allows for consistent ASCII printing through any of the hardware connections. The host print transform function can also be used to send the ASCII data stream to a printer or system by using the Send TCP/IP Spooled File (SNDTCPSPLF) command, or to an ASCII LAN attached printer.

Why use the host print transform function?

The host print transform function allows the SCS-to-ASCII or AFP-to-ASCII data stream conversion to take place on the iSeries server instead of on an emulator. Having the conversion take place on the iSeries server provides these advantages:

- Consistent output for most ASCII printers

  The host print transform function is capable of supporting many different types of ASCII data streams. For example, the Hewlett-Packard** printer control language (PCL), the IBM personal printer data stream (PPDS), and the Epson** FX and LQ data streams.

  Having the conversion done on the iSeries server ensures that the resultant ASCII data stream provides the same printed output regardless of the emulator the printer is physically attached to.

- 3812 SCS Printer Emulation

  The host print transform function is based on the 3812 SCS printer emulation of the iSeries Access for Windows work station function. Using the host print transform function, all of the ASCII printers connected to an iSeries server can perform a 3812 SCS level of function.

  Note: You cannot perform functions that your printer does not support. For example, you cannot print in 180 degree orientation when your printer only supports 0 and 90 degree orientations.

- Support for many different ASCII printers

  Without host print transform function, each emulator supports a limited number of ASCII printers. With the host print transform function, most IBM printers and a large number of other printers are supported.

- Customized printer support

  Workstation customizing objects that come with the host print transform function can be updated by the user to change or add characteristics to a particular printer. Also, if the host print transform function does not have a workstation customizing object for a printer you want to use, you can create your own.

- Support for the image object content architecture (IOCA) imbedded in an SCS data stream. Image output can be printed on PCL printers (Hewlet-Packard LaserJet**) or PPDS laser printers (IBM 4019 or 4029).

This support allows you to print the following to PCL and PPDS laser printers. This support is available to all emulators discussed in this chapter.

– OfficeVision/400 documents that contain image

Images can be brought into OfficeVision/400 documents through the Graphic Instruction.

– Incoming faxes from Facsimile Support/400

Incoming faxes saved in a folder as RFT:DCA can be printed by using the Print Document (PRTDOC) command. You can also use the Print Fax (PRTFAX) command to direct the incoming faxes directly to an ASCII print device configured to use the HPT. For V3R7, HPT has also been enhanced to do image resolution scaling which allows faxes to be printed in normal size. See the Facsimile Support for OS/400 book for more information.

• Support for conversion of double-byte SCS or AFP data stream into ASCII data stream.

For the AFP-to-ASCII data stream conversion, there are additional advantages such as support for AFP font, text, image and bar code commands.

How the host print transform function works

The host print transform function converts the print data stream just before it is sent from the iSeries server. The spooled file contains the print data and not the converted ASCII data.

Note: The host print transform function works with jobs that are sent directly to the printer (SPOOL(*NO) on the printer file).

The following figure shows the data stream origination, flow, and conversion for a spooled file when the host print transform function is enabled.

```
┌─────────────────────────────────────────────────────┐
│     AS/400  System  with  Host  Print  Transform  enabled │
└─────────────────────────────────────────────────────┘

         Application

    DDS  specifications                        Print  options

                        Printer  file

                Spooled  file  in  output  queue

                        Writer
                                    (SCS  data  stream)
                                    (AFP  data  stream)
                    Host  print  transform

                            (ASCII  data  stream  +  ATRN)
  AS/400

  Device

                        Emulator
                                    (ASCII  data  stream)
                    ASCII  printer

                                            RBAFT526-0
```
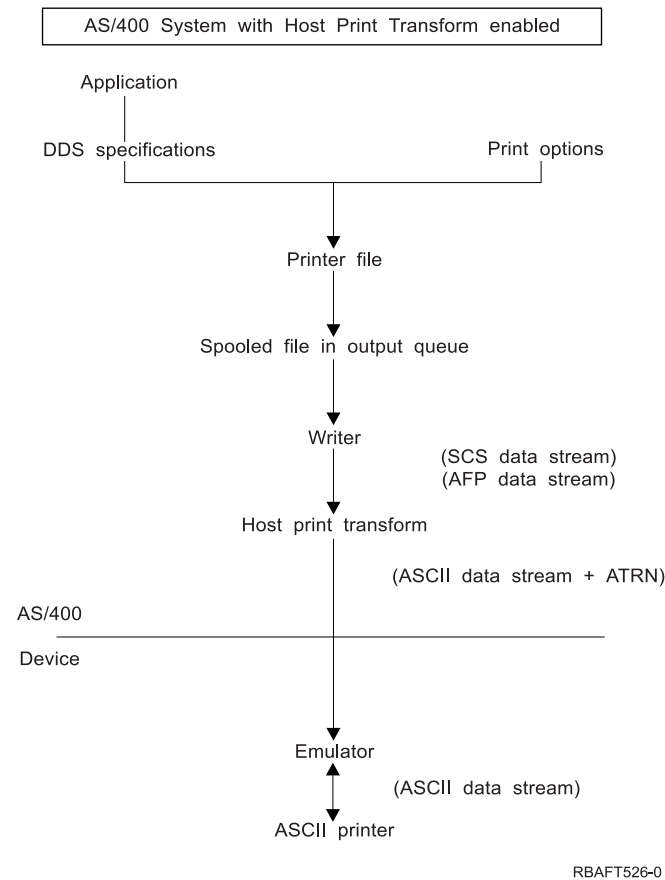
The host print transform function generates an ASCII printer data stream for a number of IBM and non-IBM printers. To generate the different ASCII data streams, the host print transform function uses iSeries server objects that describe characteristics of a particular ASCII printer. Using the SCS ASCII Transparency (ATRN) command, passes the ASCII data stream through the existing emulator. The existing emulator deletes the ASCII Transparency commands and passes the ASCII data stream that is generated by the host print transform function to the personal printer. CA/400 PC5250 and IBM Personal Communications do not support partial ASCII transparency printing.

Note: The emulator must support the SCS ASCII Transparency (ATRN) command to use it with the host print transform function. All the emulators described in this chapter support the ATRN command.

To enable the host print transform function, you work with the printer device description.

Using AFP-to-ASCII transform function

The AFP-to-ASCII transform function supports AFP font, text, bar code commands, as well as double-byte code pages and fonts. The function works well with the following types of printers:
- IBM Network printers and IBM Infoprint printers
- HP laser and ink jet printers

- IBM pages printers

On other printers, images or bar code may not be supported by AFP-to-ASCII transform function, and the text may not be positioned correctly.

Using bar codes

A bar code is a predetermined pattern of bars and spaces that represent numeric or alphanumeric information in a machine readable form. Bar codes are commonly used in many applications including, item tracking, inventory control, point of sale operations, patient care and so on.

IBM's Advanced Function Print (AFP) data stream defines an architecture for presenting bar codes. Bar codes can be printed by using printer files and the BARCODE DDS keyword. They can also be printed by using the Advanced Function Printing Utilities/400. Bar code support works best on laser printers. Dot matrix printers, and some ink jet printers do not support bar codes.

The AFP-to-ASCII transform supports the following industry standard bar code types:
- Code 39, AIM USS-39
- MSI
- UPC/CGPC Version A
- UPC/CGPC Version E
- UPC Two-digit Supplemental
- UPC Five-digit Supplemental
- EAN-8
- EAN-13
- Industrial 2-of-5
- Matrix 2-of-5
- Interleaved 2-of-5
- Interleaved 2-of-5, AIM USS-1 2/5
- Codabar 2-of-7, AIM USS-Codabar
- Code 128, AIM USS-128
- EAN Two-digit Supplemental
- EAN Five-digit Supplemental
- POSTNET
- Japan Postal
- Royal Mail
- Australian Postal 4-state
- Dutch Postal (KIX)

Limitations of AFP-to-ASCII transform function

Currently, the AFP to ASCII transform has the following limitations:
- It does not support graphics (GOCA) commands
- It does not support multi-up printing
- It does not provide support for computer output reduction (COR) printing
- It ignores the fidelity attribute of the spooled file and always performs content printing

- It can only produce 240 DPI images and 300 DPI images

Using the host print transform in raster mode

In raster mode, the Host Print Transform builds a raster image for each page of the output. The raster image is then compressed. AFP fonts must be installed on OS/400 before using the raster mode. Both the SCS-to-ASCII and AFP-to-ASCII transforms support raster mode.

Why use raster mode?

You would use raster mode for the following:
- Raster mode preserves the relative position of text, images, barcodes, and overlays if the output is positioned within the no print border of your ASCII printer.
- To transform SCS or AFP output to formats other than printer data streams. The Host Print Transform function can, for example, transform SCS or AFP output to TIFF (Tagged Image File Format) for viewing or archiving.
- To support SCS or AFP functions not available on your ASCII printer. For example, the IBM 4019 printer does not support multiple print orientations on the same page.
- To support the printing of AFP output spooled with DBCS or Unicode (UCS-2) encoded data.

Enabling raster mode

To enable raster mode:
- Install product option 8, AFP Compatibility Fonts, for Operating System/400®.
- For better font support install, IBM AFP Font Collection for IBM Operating System (5648–B45).
- For SCS or AFP output containing double byte characters (DBCS) install the Advanced Function Printing DBCS Fonts/400 (5769FN1). You can also install one of the DBCS outline font features for the IBM AFP font collection. For AFP output containing Unicode (UCS-2) encoded data, install the AFP Unicode migration fonts. For SCS output you must also specify, the OS/400 system value QIGCCDEFNT (double byte coded font).
- Configure the Host Print Transform to use a Workstation Customization Object (WSCST) with the following tag:

```
:RASTERMODE
  SCS=YES
  AFP=YES.
```

Raster mode does not support all printer data streams. Refer to the Workstation Customization Programming manual for detailed information about raster mode support.

Limitations of raster mode

The following are limitations of raster mode:
- Raster mode requires more OS/400 resources and is slower than the Host Print Transform mapping mode.
- Raster mode does not support dot matrix printers.
- The Host Print Transform can only produce raster images with 240 or 300 dots per inch (dpi). It does not support ink jet printers that have a resolution of 360 or 720 dpi.

- Raster mode does not support hardware justification of text.
- For SCS output that contains double byte characters (DBCS), you can specify only one double byte coded font. Setting the QIGCCDEFNT system value selects this font..

Enabling the host print transform function using printer device description parameters

The host print transform function is enabled by selecting certain values for parameters in the printer device description. If you need more detailed information on these parameters, see the Local Device Configuration PDF.

Parameters supporting the host print transform function

The following parameters on the printer device description are used by the host print transform function:

TRANSFORM

Host print transform function

*YES Enables the host print transform function for this printer. This function is only to be used for ASCII printers.

*NO Disables the host print transform function for this printer.

Note: When TRANSFORM(*YES) is specified, the FORMFEED parameter value for this device is ignored. The FORMFEED type is based on the value specified in the paper source 1 (PPRSRC1) parameter.

MFRTYPMDL

Manufacturer, type, and model.

Press F4 or the Help key for a list of printers supported by the host print transform function. Examples of the values for this parameter are *IBM4029 for the IBM 4029 LaserPrinter Model 10 or *HPIIID for the Hewlett-Packard LaserJet IIID**.

Note: You must choose a value for MFRTYPMDL if TRANSFORM(*YES) is specified. There is no default for this parameter. Default values for PPRSRC1, PPRSRC2, and ENVELOPE are selected based on the MFRTYPMDL value.

PPRSRC1

Paper source 1.

The value for this parameter is used to specify the size of the paper in drawer 1 or continuous size paper if applicable. Press F4 (Prompt) or the Help key for a list of paper sizes supported by the host print transform function.

Note: The paper size value is used by the host print transform function to support the computer output reduction (COR) function.

*MFRTYPMDL

This is the default. The iSeries server substitutes the value that is most common for your printer. It substitutes *LETTER for all page printers, *CONT80 for narrow-carriage continuous-feed printers, and *CONT132 for wide-carriage continuous-feed printers. If the printer uses a paper size other than the default size, you should explicitly specify that size to completely support the COR function.

PPRSRC2

Paper source 2.

The value for this parameter is used to specify the size of the paper in drawer 2. The value in PPRSRC2 is ignored for continuous feed printers. Press F4 (Prompt) or the Help key for a list of paper sizes supported by the host print transform function.

*MFRTYPMDL

This is the default. The iSeries server substitutes the value that is most common for your printer. It substitutes *LETTER for all page printers.

ENVELOPE

Envelope source.

The value for this parameter is used to specify the size of the envelope. Press F4 (Prompt) or the Help key for a list of envelope sizes supported by the host print transform function.

*MFRTYPMDL

This is the default. The iSeries server substitutes a value of *NUMBER10 if your printer supports an envelope feeder.

ASCII899

ASCII code page 899 support.

*YES Select *YES if your printer supports code page 899. Code page 899 is not resident on most ASCII printers. With the IBM 4029 LaserPrinter, a font card is required.

*NO *NO is the default. If your printer does not support code page 899, use *NO.

The following additional parameter can be used by the host print transform function when a user-defined workstation customizing object is necessary. A user-defined customizing object for the printer is usually not needed due to the extended support provided by the host print transform function.

WSCST

Workstation customizing object and library.

If the host print transform function is enabled and an object name is specified on the WSCST parameter, that object must be compatible with the host print transform function.

*NONE

The default is *NONE.

Working with printer device descriptions

The host print transform function is enabled when you specify *YES for the TRANSFORM parameter in the printer device description. The TRANSFORM parameter can be specified when the printer device description is created or when you change an existing printer device description. Because of the complexity of creating a device description manually, it is recommended that you use automatic configuration. Then, after the device description has been created, change the device description to enable the host print transform function.

Notes:

1. Automatic configuration of devices (printers, in this case) attached to the ASCII Work Station Controller is not supported. The CRTDEVPRT command must be used to create the printer device description.

2. The work station function of iSeries Access for Windows creates or changes its printer device descriptions based on the printer's session configuration. For this emulator, the host print transform function should be enabled by changing the session configuration on the personal computer and not the device description in the iSeries server.

Creating printer device descriptions using a command

You may not want your printer device descriptions to be created by automatic configuration, for example, if your printer is attached to the ASCII Work Station Controller.

In this case, you should create your printer device descriptions by typing the Create Device Description (Printer) (CRTDEVPRT) command and pressing F4 (Prompt). You will need to enter a value for most of the parameters. See "Enabling the host print transform function using printer device description parameters" on page 276 for information on the parameters and values used by the host print transform function.

Automatically creating printer device descriptions

You can have printer device descriptions created automatically if the automatic configuration (QAUTOCFG) system value is *YES. The printer device description is automatically created when:

- A display or printer is powered on
- The personal computer or PS/2 emulation programs are started

To enable the host print transform function after automatic configuration, type the Change Device Description (Printer) (CHGDEVPRT) command and press F4 (Prompt).

Note: Before you change a printer device description, it is recommended that you:
  - End the printer writer (ENDWTR command)
  - Vary off the printer device (WRKCFGSTS command)

See "Enabling the host print transform function using printer device description parameters" on page 276 for information on the parameters and values used by the host print transform function.

Note: The work station function of iSeries Access for Windows can automatically configure printers with the host print transform function enabled.

Changing an existing printer device description

When working with existing printer device descriptions, you can enable the host print transform function by changing certain parameter values in the printer device description.

To enable the host print transform function, type the Change Device Description (Printer) (CHGDEVPRT) command and press F4 (Prompt).

Note: Before you change a printer device description, it is recommended that you:
  - End the printer writer (ENDWTR command)
  - Vary off the printer device (WRKCFGSTS command)

See "Enabling the host print transform function using printer device description parameters" on page 276 for information on the parameters and values used by the host print transform function.

Note: The work station function of iSeries Access for Windows can automatically configure printers with the host print transform function enabled.

Displaying the printer device description

If you want to verify your host print transform function parameters, use the Display Device Description (DSPDEVD) command to display the printer device description. If the default value *MFRTYPMDL was specified for the PPRSRC1, PPRSRC2, and ENVELOPE parameters, the system-supplied values are shown when the device description is displayed.

Use the Change Device Description (Printer) (CHGDEVPRT) command to change any parameter values that are not correct for your printer.

Using the host print transform function with an emulator

The following emulator products provide printer emulation for ASCII printers on the iSeries server. Each section provides a brief description of the emulator, some advantages of using the host print transform function with this emulator, and suggested emulator configuration values. The host print transform function does not eliminate the need for existing emulators. Existing emulators are still necessary to attach the personal printer to the iSeries server.

Go to Page
    To Work With:

Using the host print transform function with the IBM iSeries Access for Windows work station function

The iSeries Access for Windows work station function program provides the following data link control attachment of IBM personal computer systems for work station function and printer emulation:

- Twinaxial
- Token-ring
- Asynchronous

- Synchronous

The iSeries Access for Windows work station function provides a DOS 5250 emulator for all of the supported iSeries Access for Windows Support connection methods. Up to five work station sessions, in any combination of display or printer sessions, can be configured. The work station function provides 3812 SCS printer emulation. The personal printer can be attached to the personal computer through either the parallel or serial interface.

Configuration recommendations

To use the host print transform function, change the printer session profile on the personal computer using the options on the CFGWSF configuration program. CFGWSF is the configuration program for the iSeries Access for Windows work station function. The CFGWSF program provides options for specifying:

Printer manufacturer, type, and model

Paper sizes

Envelope sizes

Symbols code page support

Workstation customizing object.

When the printer session is started, these parameters are passed from the personal computer to the iSeries server. The printer device description is created or changed to reflect the values passed from the CFGWSF program. Any changes must be made by changing the printer session profile on the personal computer instead of using the CHGDEVPRT command.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The iSeries Access for Windows work station function uses the Printer Function Table Setup (PFTSETUP) program to customize features on a printer. If you have used PFTSETUP to customize a printer, use of the host print transform function overrides this customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make a note of the printer features that you customized.

   You can use the PFTSETUP program to locate the customized features.
2. Transfer those customized features to an object compatible with the host print transform function.

   This means changing some features on this object to match the features you customized with the PFTSETUP program.
3. Use this object name as the value for the workstation customizing object parameter in that printer's session profile.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with the 3486/3487/3488 InfoWindow display

The 3486/3487/3488 InfoWindow II display stations are twinaxial display stations that attach to the iSeries server using a twinaxial controller or a remote workstation controller. The displays support up to two display sessions and a printer session. The 3486/3487/3488 display provides 5256, 4214, or 3812 SCS printer emulation for many IBM personal printers and some Hewlett-Packard personal printers.

Configuration recommendations

Configure printers that are supported by the InfoWindow display as recommended in the InfoWindow display's configuration menus. Configure any printer that is not included in the list of supported printers as a 4201/4202 printer.

When you leave the InfoWindow display's configuration menu, the printer device description is automatically created if both the following are true:

- The printer is powered on
- Automatic configuration is on

After the printer device description is created on OS/400, enable the host print transform function using the CHGDEVPRT command. The display's printer configuration is not used after the host print transform function is enabled. The data stream generated for the printer is based on the MFRTYPMDL parameter value specified in the printer device description on OS/400.

Note: Do not change the display's printer configuration after your device description has been created on the iSeries server. Doing so can cause the OS/400 printer device description to be replaced. In this case, the host print transform function is no longer enabled. The CHGDEVPRT command can be used to enable the host print transform function again.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The 348x displays can use a workstation customizing object (located in the device description of the display) to define the printer attached to it. Conversely, the host print transform function uses a workstation customizing object located in the device description of the printer. If you have customized some printer features in the display's workstation customizing object, use of the host print transform function overrides the customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.

   The TRANSFORM parameter value for the printer device description must be *YES.

2. Use a workstation customizing object that is compatible with the host print transform function.

   Customize the printer features in this object to match the customized features that you used in the display's workstation customizing object.

3. Use the workstation customizing object name that is compatible with the host print transform function as the value for the WSCST parameter in the printer device description.

> Remember: The location of the WSCST object name (in the printer device description, not the display device description) is important when using customization with the host print transform function.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with the 3477 InfoWindow display

The 3477 InfoWindow display station is a twinaxial display station that attaches to the iSeries server using a twinaxial controller or a remote workstation controller such as 5294, 5394, or 5494. The display supports up to two display sessions, or one display session and one printer session. The 3477 display provides 5256, 4214, or 5219 SCS printer emulation for most IBM personal printers.

Configuration recommendations

Configure printers that are supported by the InfoWindow display as recommended in Appendix A of the *IBM 3477 InfoWindow user's guide* Configure printers that are not supported by the 3477 with the following values:

- Type A for *Printer Character Set*
- 5256 for *Printer Emulation*

When you leave the display's configuration menu, the printer device description is automatically created if both the following are true:

- The printer is powered on
- Automatic configuration is on

After the printer device description is created on the iSeries server, enable the host print transform function using the CHGDEVPRT command. The display's printer configuration is not used after the host print transform function is enabled. The data stream generated for the printer is based on the MFRTYPMDL parameter value specified in the printer device description on the iSeries server.

> Note: Do not change the display's printer configuration after your device description has been created on the iSeries server. Doing so can cause the OS/400 printer device description to be replaced. In this case, the host print transform function is no longer enabled. The CHGDEVPRT command can be used to enable the host print transform function again.

Understanding the 3477 power-on initialization sequence

The 3477 sends a power-on initialization sequence to the attached printer every time the printer or display is powered on. This initialization is designed for IBM printers. If a printer is attached that does not support the IBM data stream, some unrecognizable characters may be printed. After this initialization sequence, the data stream generated is based on the MFRTYPMDL parameter in the printer device description on the iSeries server. No more unrecognizable characters will be printed.

If your 3477 is a Model H, you can turn off the power-on initialization sequence by doing the following:

1. From the 3477's setup menu, select *Test Workstation*.
2. Press the Alt, Shift, and Setup keys together.
3. The F6 key is now set to switch between power-on initialization active and power-on initialization not active.

   Pressing F6 displays +6 or *6 on the bottom of the screen. If +6 is displayed, power-on initialization is not active. If *6 is displayed, power-on initialization is active.
4. Press the Reset key to return to the 3477 setup menu.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The 3477 Model H is the only 3477 InfoWindow display that supports customization. The 3477 Model H display can use a workstation customizing object (located in the device description of the display) to define the printer attached to it. Conversely, the host print transform function uses a workstation customizing object located in the device description of the printer. If you have customized some printer features in the display's workstation object, use of the host print transform function overrides the customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.

   The TRANSFORM parameter value for the printer device description must be *YES.
2. Use a workstation customizing object that is compatible with the host print transform function.

   Customize the printer features in this object to match the customized features that you used in the display's workstation customizing object.
3. Use the workstation customizing object name that is compatible with the host print transform function as the value for the WSCST parameter in the printer device description.

   Remember: The location of the WSCST object name (in the printer device description, <u>not</u> the display device description) is important when using customization with the host print transform function.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with the 3197 display station

The 3197 display station is a twinaxial display station that attaches to the iSeries server using a twinaxial controller or a remote workstation controller such as 5294, 5394, or 5494. The 3197 supports a single display, two displays, or one display and one emulated printer. The 3197 provides 5256 or 4214 printer emulation for many IBM personal printers.

Configuration recommendations

Refer to the IBM *3197 Display Stations User's Guide* to set up your display for a printer session. Select *Display-Printer Mode* with a *Printer Device ID* of 5256. Specify 850 as the *Printer Character Set Selection*.

When you leave the InfoWindow display's configuration menu, the printer device description is automatically created if both the following are true:

- The printer is powered on
- Automatic configuration is on

After the printer device description is created on the iSeries server, enable the host print transform function using the CHGDEVPRT command. The display's printer configuration is not used after the host print transform function is enabled. The data stream generated for the printer is based on the MFRTYPMDL parameter value specified in the printer device description on OS/400.

Notes:

1. The 3197 sends a power-on initialization sequence to the attached printer every time the printer or display is powered on. This initialization is designed for IBM printers. If a printer is attached that does not support the IBM data stream, some unrecognizable characters may be printed. After this initialization sequence, the data stream generated is based on the MFRTYPMDL parameter in the printer device description on OS/400. No more unrecognizable characters will be printed.

2. Do not change the display's printer configuration after your device description has been created on the iSeries server. Doing so can cause the OS/400 printer device description to be replaced. In this case, the host print transform function is no longer enabled. The CHGDEVPRT command can be used to enable the host print transform function again.

The order in which the 3197 display and its attached printer are powered on can affect the iSeries server's ability to recognize the printer. This order is dependent on the model of the 3197. Use the Work with Configuration Status (WRKCFGSTS) command to check the status of a printer device. If the status of the printer is *vary on pending* or if the printer is not automatically configured, reverse the order in which the display and printer were powered on.

Using the host print transform function with the ASCII work station controller

The ASCII Work Station Controller (AWSC) resides on the iSeries server and allows up to 18 ASCII displays or printers to attach to each controller. The AWSC provides 3812, 5219, 5224, or 5256 SCS printer emulation for most IBM printers that support an EIA-232 or EIA-422 serial interface.

Configuration recommendations

There is no automatic configuration capability for printers attached to the AWSC. The Create Device Description (Printer) (CRTDEVPRT) command must be used to create the configuration descriptions. For existing printers, use the Change Device Description (Printer) (CHGDEVPRT) command.

Configure printers not supported by the AWSC with 4019 as the device type, Model 1 as the model, and 3812 as the emulated printer. Many of the other parameters depend on the attached printer (line speed, data bits, and so on). To

avoid OS/400 parameter checking, specify *YES as the value on the emulating ASCII parameter in the printer device description. For example, this allows a communications line speed of 38 400 bps.

After the host print transform function is enabled, the device type and model parameters in the device description have no effect on the data stream sent to the printer. The data stream generated for the printer is based on the MFRTYPMDL parameter in the OS/400 printer device description.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The ASCII work station controller can use a workstation customizing object in the printer device description to define the printer's characteristics. That workstation customizing object can be customized to use special features of a printer that the AWSC does not support.

If the host print transform function is enabled, the customizing object specified for the AWSC printer device description must be removed or replaced. The customizing objects created for the AWSC are not compatible with the host print transform function.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.

   The TRANSFORM parameter value for the printer device description must be *YES.

2. Use a workstation customizing object that is compatible with the host print transform function.

   Customize the printer features in this object to match the customized features that you initially used in the workstation customizing object specified in the AWSC printer device description.

3. Use the workstation customizing object name that is compatible with the host print transform function as the new value for the WSCST parameter in the printer device description.

   Remember: The AWSC and the host print transform function both use the workstation customizing object parameter in the printer device description. That object must contain the customized features and be compatible with the host print transform function.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with OS/2 5250 work station feature

The OS/2 5250 Work Station Feature is part of OS/2 Extended Edition or Extended Services Communications Manager. Communications Manager provides twinaxial, IBM token-ring, X.25, asynchronous, and synchronous data link control attachment of personal computers and PS/2 personal computer systems to the iSeries server.

5250 Work Station Feature is a 5250 emulator for all the connection types supported by the Communications Manager. Five work station sessions in any combination of display or printer sessions can be configured. The 5250 Work Station Feature provides 5219, 5224, and 5256 printer emulation for personal printers. The personal printer can be attached to the personal computer through either the parallel or serial interface.

Configuration recommendations

OS/2 5250 Work Station Feature can support printers through a printer function table (PFT). To use PFTs to create your printer device descriptions on the iSeries server, follow these steps:

1. If you are working with a printer that has a PFT, you need to:
   a. Select printer emulation type 5219, 5224, or 5256.
   b. Select the PFT for the printer you are working with.
   c. Go to step 3.
2. If you are working with a printer that does not have a PFT, you need to:
   a. Copy the DEFAULT.PFT file from the QIWSFLR or QIWSFL2 folder on the iSeries server into your personal computer's CMLIB directory. Name the file PFTHPT.
   b. Configure the printer session on your personal computer with 5219 printer emulation and use the PFT named PFTHPT.
   c. Go to step 3.
3. Start your printer session on the personal computer. This creates a new device description on the iSeries server.
4. Use the CHGDEVPRT command to enable the host print transform function. See "Enabling the host print transform function using printer device description parameters" on page 276 for the parameters that enable the host print transform function.
5. On the personal computer, stop the emulated print session and then restart it to reestablish the printer session. The session ended when the printer device was varied off (at the iSeries server) before you used the CHGDEVPRT command.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The OS/2 5250 Work Station Feature supports printers through a printer function table (PFT). If you have customized a printer by changing the supplied PFT, use of the host print transform function overrides this customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.
   The TRANSFORM parameter value of the printer device description must be *YES.
2. Make a note of the features you customized in the PFT.
3. Use a workstation customizing object that is compatible with the host print transform function.
   Update the features in this object to reflect the customized features you used in the PFT.

4. Use the workstation customizing object name that is compatible with the host print transform function as the value for the WSCST parameter in the printer device description.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with OS/2 5250 emulation

OS/2 5250 Emulation is part of OS/2 Communications Manager/2. Communications Manager/2 provides coaxial, ISDN, LAN, SDLC, twinaxial, and X.25 attachment of IBM PS/2 and other work stations to the iSeries server.

OS/2 5250 Emulation is a 5250 emulator for all the connection types supported by Communications Manager/2. Up to 15 work station sessions in any combination of display or printer sessions can be configured. OS/2 5250 Emulation provides 3812 SCS printer emulation for all printers supported by OS/2 2.0 or later. Personal printers can be attached to personal computers through either the parallel or serial interface.

Configuration recommendations

Configure printers using an OS/2 printer driver. If the printer driver required by your personal printer is not available, select a printer driver that most closely matches the one your printer uses.

When the OS/2 5250 printer session has been connected, the iSeries server automatically creates a printer device description.

On the iSeries server, use the CHGDEVPRT command to enable the host print transform function. See "Enabling the host print transform function using printer device description parameters" on page 276 for the parameters that enable the host print transform function.

On the work station, restart the printer session. You need to do this to reestablish the printer session. The session ended when the printer device was varied off (at the iSeries server) before you used the CHGDEVPRT command.

Using the host print transform function with the RUMBA/400 program

The iSeries Access for Windows program provides the following data link control attachment of IBM personal computer systems for work station function and printer emulation:

- Twinaxial
- Token-ring
- Asynchronous
- Synchronous

RUMBA/400 provides an OS/2 or Windows 5250 emulator for all of the supported connection methods. Any number of work station sessions (limited by available memory) in any combination of display or printer sessions can be configured. RUMBA/400 provides 3812 SCS printer emulation for all printers supported by OS/2 2.0 or later and Microsoft Windows 3.0 or later. The personal printers can be attached to personal computers through either the parallel or serial interface.

Configuration recommendations

Configure printers using an OS/2 or Windows printer driver. If the printer driver required by your printer is not available, select a printer driver that most closely matches the one your printer uses.

When the RUMBA/400 printer session has been connected, the iSeries server automatically creates a printer device description.

On the iSeries server, use the CHGDEVPRT command to enable the host print transform function. See "Enabling the host print transform function using printer device description parameters" on page 276 for the parameters that enable the host print transform function.

On the personal computer, restart the printer session. You need to do this to reestablish the printer session. The session ended when the printer device was varied off (at the iSeries server) before you used the CHGDEVPRT command.

Using the host print transform function with the IBM enhanced 5250 or the IBM S36/38 work station emulation program

The Enhanced 5250 Emulation Program is a personal computer program. This program allows a non-Micro Channel** personal computer to emulate a display, two displays, or a display and a printer session on the iSeries server. The program uses a card to communicate through a twinaxial cable to the twinaxial port or remote controller attached to the iSeries server.

The S36/38 Work Station Emulation Program is a personal computer program. This program allows a Micro Channel® PS/2 to emulate up to four display or printer sessions. The program uses a card in the PS/2 to communicate through a twinaxial cable to a twinaxial port or remote controller.

Configuration recommendations

Configure printers supported by the IBM Enhanced 5250 Emulation program or S36/38 Work Station Emulation program as recommended in the *IBM Enhanced 5250 Emulation Program User's Guide* , G570-2221, or the *IBM S36/38 Work Station Emulation User's Guide* SC21-8068. For example, use the IBM4019.PFT and select 5219 Printer emulation for the IBM 4019 LaserPrinter. Configure printers that were not previously supported by the emulation program using the DEFAULT.PFT file and selecting 5219 printer emulation.

When the printer session is started, the iSeries server automatically creates the printer device description (if automatic configuration is turned on).

On the iSeries server, use the CHGDEVPRT command to enable the host print transform function. See "Enabling the host print transform function using printer device description parameters" on page 276 for the parameters that enable the host print transform function.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not need to create a customizing object for the printer due to the extended support provided by the host print transform function.

The Enhanced 5250 and S36/38 Work Station Emulation programs support printers through a printer function table (PFT). If you have customized a printer by changing the supplied PFT, use of the host print transform function overrides this customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.

   The TRANSFORM parameter value for the printer device description must be *YES.

2. Make a note of the features you customized in the PFT.

3. Use a workstation customizing object that is compatible with the host print transform function.

   Update the features in this object to reflect the customized features you used in the PFT.

4. Use the workstation customizing object name that is compatible with the host print transform function as the value for the WSCST parameter in the printer device description.

See the Workstation Customization Programming manual for detailed information on customizing printers.

Using the host print transform function with the IBM remote 5250 emulation program

The Remote 5250 Emulation Program is a personal computer program that allows a personal computer or PS/2 to emulate an IBM 5294 Remote Workstation Controller. The Remote 5250 Emulation Program relies on an SDLC communications line to communicate with the iSeries server. The Remote 5250 Emulation Program provides the ability to establish up to two display sessions or one display and one printer session.

Configuration recommendations

See the *Remote 5250 Emulation Program User's Guide* , G570-2203, for information on configuring printer sessions. For example, use the IBM5204.PDT file and select 5219 Printer emulation for the IBM 5204 Quickwriter*. Configure printers that were not previously supported by the emulation program using the TBLPRT.PDT and selecting 5219 printer emulation.

When the printer session is started, the iSeries server automatically creates the printer device description (if automatic configuration is turned on).

On the iSeries server, use the CHGDEVPRT command to enable the host print transform function. See "Enabling the host print transform function using printer device description parameters" on page 276 for the parameters that enable the host print transform function.

Maintaining printer customization

Before you do the work of creating a workstation customizing object for a printer, try printing jobs using the host print transform function support. You may not have to create a customizing object for the printer due to the extended support provided by the host print transform function.

The Remote 5250 Emulation Program supports printers through a printer description table (PDT). If you have customized a printer by changing the supplied PDT, use of the host print transform function overrides this customization.

However, if you need to customize your printer while using the host print transform function, you should:

1. Make sure the host print transform function is enabled.

   The TRANSFORM parameter value for the printer device description must be *YES.

2. Make a note of the features you customized in the PDT.

3. Use a workstation customizing object that is compatible with the host print transform function.

   Update the features in this object to reflect the customized features you used in the PDT.

4. Use the workstation customizing object name that is compatible with the host print transform function as the value for the WSCST parameter in the printer device description.

See the Workstation Customization programming manual for detailed information on customizing printers.

Chapter 14. Working with the image print transform function

This chapter provides information about the image print transform function and how to enable it to provide additional support for printers that are attached to an iSeries server. The image print transform function is an OS/400 function that is capable of converting image or PostScript** data streams into various ASCII and non-ASCII printer data streams. The conversion takes place on the iSeries server, which means the data stream generated is independent of any printer emulators or hardware connections.

What is the image print transform function?

The image print transform function converts image or print data from one format into another. A supporting printer interprets the resultant printer data stream. The image print transform function can convert the following data streams:

- Tag Image File Format** (TIFF)
- Graphics Interchange Format** (GIF)
- OS/2 and Windows** Bitmap (BMP)
- PostScript** Level 1 with DBCS extensions

The image print transform function can generate the following printer data streams:

- Advanced Function Print Data Stream (AFPDS)
- Hewlett-Packard** Printer Control Language (PCL)
- PostScript** Level 1

Similar to the host print transform function, the image print transform function converts the data on the iSeries server instead of an emulator.

When a data stream is converted by the image print transform function, the printer data stream that is created contains a bit-mapped image. A *bit-mapped image* is an array of numerical values. Each value represents part or all of a *pixel*. A *pixel* is a single point or dot of an image. An image is usually measured in terms of pixels, for both width and height. The *resolution* of an image is then defined as the number of pixels (dots) per unit of measure. For example, a resolution supported by many printers is 300 dots per inch (dpi). Therefore, an image having dimensions 1200 pixels by 1500 pixels would have a width of 4 inches and a height of 5 inches when printed at 300 dpi.

Why use the image print transform function?

There are many advantages for using the image print transform function.

- Support for Intelligent Printer Data Stream (IPDS) printers

  TIFF, GIF, and BMP image files, as well as PostScript Level 1 files, can be converted to AFPDS format and printed on IPDS printers configured AFP(*YES).

- Support for ASCII printers

  TIFF, GIF, and BMP image files, as well as PostScript Level 1 files, can be converted to PCL-5 and PostScript Level 1 format and printed on ASCII printers supporting these languages.

Note: You cannot convert PostScript to PostScript using the image print transform function. When the input and output data streams are PostScript, the data is sent directly to the output destination without conversion.

- Customized printer support

  Image configuration objects are used with the image print transform function to specify certain characteristics of the converted data streams. When associated with the device description information for a printer connected to an iSeries server, an image configuration object acts as a template for the converted data stream. Attributes such as data stream format, color, and resolution are all specified in the image configuration object.

- Additional capabilities

  In addition to converting data from one format to another, other functions can be performed by the image print transform function. Among these are the ability to reduce color, compress data, and change photometricity. For more information about the features of the image print transform function, consult the System API Reference topic in the iSeries Information Center.

  Note: You cannot perform functions that your printer does not support. For example, you cannot print in landscape orientation when your printer only supports portrait orientation.

Printing with image print transform function

The image print transform function works with ASCII printers as well as IPDS printers that have AFP(*YES) specified in the configuration. When the image print transform function is used, the transform doesn't take place until after the data stream is spooled. Then, when the spooled file is printed or sent to a remote output queue, it is first sent to the image print transform function to be transformed. Printing with the image print transform function is done automatically when:

- A printer device is created with the host printer transform function enabled.
- The device or remote output queue has an image configuration object defined.

For more information about creating a device with HPT enabled, see Chapter 13, "Working with the host print transform function" on page 271.

Printing to an ASCII printer

To enable the image print transform function when printing to an ASCII printer, do the following:

- Ensure that the spooled file is a *USERASCII spooled file.
- Verify that the printer device description has the TRANSFORM field set to *YES.
- Verify that the printer device description has the IMGCFG field set to a valid value other than *NONE.

The TRANSFORM field and the IMGCFG field can be set when the device description is created with the Create Device Desc (Printer) (CRTDEVPRT) command, or changed after the device description was created with the Change Device Desc (Printer) (CHGDEVPRT) command.

Printing to an IPDS printer

To enable the image print transform function when printing to an IPDS printer that has AFP(*YES) specified in the configuration, do the following:

- Ensure that the spooled file is a *USERASCII spooled file.
- Verify that the printer device description has the IMGCFG field set to a valid value other than *NONE.

The IMGCFG field can be set either when the device description is created with the Create Device Desc (Printer) (CRTDEVPRT) command, or changed after the device description was created with Change Device Desc (printer) (CHGDEVPRT) command.

Printing with remote output queues

To enable the image print transform function when printing or sending spooled files to a remote output queue, do the following:

- Ensure that the spooled file is a *USERASCII spooled file.
- Verify that the output queue has the TRANSFORM field set to *YES.
- Verify that the output queue has the IMGCFG field set to a valid value other than *NONE.

You can set the TRANSFORM field and the IMGCFG field when you create the output queue with the Create Output Queue (CRTOUTQ) command. You can change both fields after creating the output queue with the Change Output Queue (CHGOUTQ) command.

How output attributes are derived

The following output attributes are derived from the image configuration object.

- Data stream format
- Photometric interpretation
- Resolution units
- Horizontal resolution
- Vertical resolution
- Compression type
- Bits per sample
- No print borders (left, right, top, bottom)

Note: If any of the above attributes are specified in the user defined data attribute of the spooled file with the convert image API, then only those attributes that are not specified will be derived from the image configuration object.

The output paper size attribute is derived from the printer device description if the output data stream format is AFPDS and the printer is an IPDS printer that has AFP(*YES) specified in the configuration.

The Paper Size attribute is derived from the printer file (i.e. spooled file attributes) if the output data stream format is not AFPDS and the printer is not an IPDS printer that has AFP(*YES) specified in the configuration.

Determining if input data stream is in final form

You can use the output data stream format to determine if an input data stream is in final form. Once the input data stream is in final form, no transform is needed. If the input data stream format matches the output data stream format, all other output attributes are ignored. No conversion is done and the spooled file is sent as is. Currently, this is only available for a PostScript data stream.

Printing with convert image API

The convert image API provides the same transform capabilities as the image print transform function. In addition, printing with the convert image API gives the user more control over how the output will look than the image print transform function does. It gives the user the ability to immediately transform a data stream when delaying the transform is not desired. It also has more options regarding the type of input object and output object. The convert image API supports input from and output to an Integrated File System (IFS) file, a spooled file or main storage. The convert image API can also generate a spooled file which can then be transformed with the image print transform function. When this is done, the convert image API stores all the values needed to do the transform in the user defined data attribute of the spooled file for later use by the image print transform function. For more information on how to use the convert image API, see the System API Reference topic in the iSeries Information Center.

Image configuration objects

An image configuration object contains various printer characteristics that the image print transform function and the convert image API use when creating output. An image configuration object is a list of characteristics that is supported by the printer it represents, acting as a template which guides the transform process. Each image configuration object has values for the following:

- Data stream format
- Photometric interpretation
- Resolution units
- Horizontal resolution
- Vertical resolution
- Compression type
- Bits per sample
- No-print borders (left, right, top, bottom)

All of these fields can be overridden by using the convert image API and specifying a value for the field of the same name.

Special values of image configurations

The following values are allowed for the image configuration (IMGCFG) field of the CRTDEVPRT, CHGDEVPRT, CRTOUTQ and CHGOUTQ commands. You can also use these values when using the convert image API. For more information on how to use these values, see the System API Reference book. Each value is described in terms of the data streams that are supported, the maximum resolution in dots per inch (dpi), and whether the printer has color or supports compression.

Printers supporting PCL data streams

*IMGA01
>   PCL 300-dpi printer

*IMGA02
>   PCL 600-dpi printer

*IMGA03
>   PCL 1200-dpi printer

*IMGA04

     PCL 300-dpi color printer

*IMGA05

     PCL 600-dpi color printer

*IMGA06

     PCL 1200-dpi color printer

*IMGA07

     PCL 75-dpi printer (No compression)

*IMGA08

     PCL 600-dpi color printer with larger no-print border

*IMGA09

     PCL 300-dpi printer (No compression)

Printers supporting PostScript data streams

*IMGB01

     PostScript 300-dpi printer

*IMGB02

     PostScript 600-dpi printer

*IMGB03

     PostScript 1200-dpi printer

*IMGB04

     PostScript 300-dpi color printer

*IMGB05

     PostScript 600-dpi color printer

*IMGB06

     PostScript 1200-dpi color printer

*IMGB07

     PostScript 600x300-dpi color printer

*IMGB08

     PostScript 1200x300-dpi color printer

*IMGB09

     PostScript 360-dpi color printer

*IMGB10

     PostScript 720-dpi color printer

*IMGB11

     PostScript 1440x720-dpi color printer

*IMGB12

     PostScript 400-dpi printer

*IMGB13

     PostScript 800-dpi color printer

*IMGB14

     PostScript 600-dpi color printer with larger no-print border

*IMGB15

     PostScript 300-dpi color printer with larger no-print border

Printers supporting IPDS data streams

*IMGC01
>   IPDS 240-dpi printer

*IMGC02
>   IPDS 300-dpi printer

*IMGC03
>   IPDS 600-dpi printer

*IMGC04
>   IPDS 1200-dpi printer

*IMGC05
>   IPDS 240-dpi printer with no-print border

*IMGC06
>   IPDS 300-dpi printer with no-print border

*IMGC07
>   IPDS 600-dpi printer with no-print border

*IMGC08
>   IPDS 1200-dpi printer with no-print border

*IMGC09
>   IPDS 240-dpi printer (IM/1 image)

*IMGC10
>   IPDS 240-dpi printer (IM/1 image) with no-print border

*IMGC11
>   IPDS 240-dpi printer (CCITT G4)

Printers supporting PCL and PostScript data streams

*IMGD01
>   PCL/PostScript 300-dpi printer

*IMGD02
>   PCL/PostScript 600-dpi printer

*IMGD03
>   PCL/PostScript 1200-dpi printer

*IMGD04
>   PCL/PostScript 300-dpi color printer

*IMGD05
>   PCL/PostScript 600-dpi color printer

*IMGD06
>   PCL/PostScript 1200-dpi color printer

*IMGD07
>   PCL 300-dpi/PostScript 600-dpi printer

*IMGD08
>   PCL 300-dpi/PostScript 1200-dpi printer

*IMGD09
>   PCL 600-dpi/PostScript 300-dpi printer

*IMGD10
>   PCL 600-dpi/PostScript 1200-dpi printer

*IMGD11

 PCL/PostScript 600-dpi color printer with larger no-print border

The recommended image configuration objects are listed below for some common printers.

*IMGD01

 Compaq Pagemarc 20

*IMGA01

 Epson EPCL-4 Printer

*IMGA02

 Epson EPCL-5 Printer

*IMGB10

 Epson Stylus Photo with PostScript

*IMGB11

 Epson Stylus Color 600, 800 with PostScript

*IMGA04

 HP Color Laserjet 5

*IMGD04

 HP Color Laserjet 5M

*IMGA04

 HP Deskjet 560C, 820C, 1200C

*IMGA01

 HP Deskjet 500, 600, 1200

*IMGA04

 HP Deskjet 1600C, 1600CN

*IMGD04

 HP Deskjet 1600CM

*IMGA09

 HP Laserjet II, IID, IIP

*IMGB01

 HP Laserjet II, IID, IIP with PostScript

*IMGA01

 HP Laserjet III, IIID, IIISi, 4L

*IMGD01

 HP Laserjet III, IIID, IIISi, 4L with PostScript

*IMGA02

 HP Laserjet 4, 4P, 4V, 4Si, 4 Plus

*IMGD02

 HP Laserjet 4M, 4MP, 4MV, 4Si MX, 4M Plus

*IMGA02

 HP Laserjet 5, 5P, 5Si

*IMGD02

 HP Laserjet 5M, 5MP, 5Si MX

*IMGA02

 HP Laserjet 6, 6P, 6L

*IMGD02
>   HP Laserjet 6M, 6MP

*IMGD02
>   IBM 3112, 3116 Page Printer with IPDS feature

*IMGA02
>   IBM 3112, 3116 Page Printer (ASCII/LAN)

*IMGD02
>   IBM 3112, 3116 Page Printer with PostScript

*IMGC01
>   IBM 3130, 3160-1 AF Printer (240-pel mode)

*IMGC02
>   IBM 3130 AF Printer (300-pel mode)

*IMGC09
>   IBM 3825, 3827, 3828 AF Printer

*IMGC01
>   IBM 3825, 3827, 3828 AF Printer (with AFIG)

*IMGC01
>   IBM 3829 AF Printer

*IMGC10
>   IBM 3835-001 AF Printer

*IMGC05
>   IBM 3835-001 AF Printer (with AFIG)

*IMGC05
>   IBM 3835-002, 3900 AF Printer

*IMGA01
>   IBM 3912, 3916 Page Printer (ASCII/LAN)

*IMGC06
>   IBM 3912, 3916 Page Printer with IPDS feature (twinax)

*IMGA01
>   IBM 3930-03 Page Printer

*IMGD01
>   IBM 3930-03 Page Printer with PostScript

*IMGC02
>   IBM 3935 AF Printer

*IMGA09
>   IBM 4019 LaserPrinters (HP mode)

*IMGB01
>   IBM 4019 LaserPrinters with PostScript

*IMGC06
>   IBM 4028 LaserPrinters

*IMGA01
>   IBM 4029 LaserPrinters

*IMGB02
>   IBM 4029 LaserPrinters with PostScript

*IMGA01
>IBM 4039 LaserPrinters

*IMGD07
>IBM 4039 LaserPrinters with PostScript

*IMGA02
>IBM 4049 LaserPrinters

*IMGD02
>IBM 4049 LaserPrinters with PostScript

*IMGB09
>IBM 4079 Color Jetprinter PS

*IMGB05
>IBM 4303 Network Color Printer

*IMGC06
>IBM 4312, 4317, 4324 NP with IPDS feature (twinax)

*IMGC06
>IBM 4312, 4317, 4324 NP with IPDS feature (LAN)

*IMGA02
>IBM 4312, 4317, 4324 NP (ASCII/LAN)

*IMGD02
>IBM 4312, 4317, 4324 NP with PostScript (ASCII/LAN)

*IMGC03
>IBM InfoPrint 60

*IMGC05
>IBM InfoPrint 62 Model 2

*IMGC06
>IBM InfoPrint 62 Model 3

*IMGB05
>IBM InfoColor® 70

*IMGC05
>IBM InfoPrint 4000

*IMGC06
>IBM InfoPrint 4000 High Resolution

*IMGB02
>Lexmark 4039Plus

*IMGD11
>Lexmark Optra C Color Printer

*IMGA02
>Lexmark Optra E, E+

*IMGD02
>Lexmark Optra N

*IMGD02
>Lexmark Optra R+, Rx+, Lx+, Lxn+

*IMGD02
>Lexmark Optra S Printers

*IMGD05

Lexmark Optra SC Color Printer

*IMGA01

Okidata OL400 LED Page Printer

*IMGA02

Okidata OL800, OL810 LED Page Printers

*IMGB12

QMS 2025, 3225

*IMGD04

QMS Magicolor CX

*IMGB09

Tektronix Phaser 140

*IMGB04

Tektronix Phaser 300

*IMGB05

Tektronix Phaser 400

*IMGB05

Tektronix Phaser 540, 550

*IMGB06

Tektronix Phaser 560

*IMGA01

Xerox 4219/MRP

*IMGA02

Xerox 4220/MRP

*IMGA02

Xerox 4230 DocuPrinter

*IMGA02

Xerox 4512, 4517 Network Printer

*IMGB13

Xerox 4520mp Printer

*IMGD04

Xerox 4700 II Color Document Printer

*IMGB08

Xerox 4915 Color Laser Printer

*IMGB05

Xerox 4920, 4925 Color Laser Printer

Converting postscript data streams

Converting PostScript data streams is performed differently from converting image data streams. PostScript conversion may require font files in order to rasterize the data. You can also find more debugging and message information if the PostScript file does not convert correctly.

Fonts

In order to convert PostScript files effectively, PostScript fonts are required to convert text and symbols into bit-mapped images. Below are lists of fonts supplied

by IBM for use with the image print transform function. Each set of fonts is located in the IFS in the specified directory. For each font name, there is a corresponding font file containing rasterization information. This mapping information is stored in the *psfonts.map* file.

Note: DO NOT alter the IBM supplied font files or the psfonts.map file shipped with OS/400. Changing a font file or font mapping can cause unpredictable results.

Table 26 shows the Latin Fonts that are stored in the following directory:

`/QIBM/ProdData/OS400/Fonts/PSFonts/Latin`

Table 26. Latin Fonts.

Font	Associated file
TimesNewRoman	TNR.PFB
TimesNewRoman-Bold	TNRB.PFB
TimesNewRoman-BoldItalic	TNRBI.PFB
TimesNewRoman-Italic	TNRI.PFB
Helvetica	HEL.PFB
Helvetica-Bold	HELB.PFB
Helvetica-BoldItalic	HELBI.PFB
Helvetica-Italic	HELI.PFB
Courier	COU.PFB
Courier-Bold	COUB.PFB
Courier-BoldItalic	COUBI.PFB
Courier-Italic	COUI.PFB

Table 27 shows the symbol fonts that are stored in the following directory:

`/QIBM/ProdData/OS400/Fonts/PSFonts/Symbols`

Table 27. Symbol Fonts.

Font	Associated file
CourierSymbols	COU_S.PFB
CourierSymbols-Bold	COU_SB.PFB
HelveticaSymbols	HEL_S.PFB
HelveticaSymbols-Bold	HEL_SB.PFB
TimesNewRomanSymbols	TNR_S.PFB
TimesNewRomanSymbols-Bold	TNR_SB.PFB

User supplied fonts

To enhance the capabilities of the image print transform function when converting PostScript data streams, you can add your own font files to be used in conjunction with the IBM supplied fonts shipped with OS/400. These fonts are called user supplied fonts. The user supplied font mapping file, *psfonts.map*, is stored in the directory as `/QIBM/UserData/OS400/Fonts`.

It behaves the same way as the psfonts.map file that is shipped with OS/400. An important difference is that the image print transform function will search for fonts by looking first at the user supplied font mapping file, and then at the OS/400 font mapping file.

To add a user supplied font, do the following:

1. Use an ASCII text editor to open the *psfonts.map* file located in /QIBM/UserData/OS400/Fonts. If this file does not exist, you need to create it.
2. Add a new line to the file to include the new font name AND associated path and file name. For example:

   ```
   font MyNewFont    /QIBM/UserData/OS400/Fonts/PSFonts/MNF.PFB
   ```

   where MyNewFont is the name of the font, and MNF.PFB is the associated font file.
3. Save the new *psfonts.map file*.
4. Copy the font file into the directory specified in the line added to the *psfonts.map* file.

To delete a user supplied font, simply remove the line mapping the font name to its associated file in the *psfonts.map* file, and remove the font file from OS/400.

Font substitutions

When a font requested within a PostScript data stream is not available on OS/400, a font substitution can be defined if there is a similar font available. A font substitution is the mapping of a font name to a font that is available and very similar (in terms of its rasterization properties) to the font file being replaced. You can also specify a font substitution if an existing font mapping is producing undesirable output.

Table 28 shows the font substitution mappings that are defined for Latin Fonts.

Table 28. Latin Fonts.

Font	Associated file
Times-Roman	TNR.PFB
Times-Bold	TNRB.PFB
Times-BoldItalic	TNRBI.PFB
Times-Italic	TNRI.PFB
Helvetica-BoldOblique	HELBI.PFB
Helvetica-Oblique	HELI.PFB
Courier-BoldOblique	COUBI.PFB
Courier-Oblique	COUI.PFB

The following font symbol substitution is also defined.

Table 29. Latin Fonts.

Font	Associated file
Symbol	TNR_S.PFB

To define a font substitution, do the following:

1. Use an ASCII text editor to open the *psfonts.map* file that is located in
   `QIBM/UserData/OS400/Fonts`

   If this file does not exist, you need to create it.
2. Add a new line to the file to include the font name AND the path and file
   name of the font file you want to use as a substitute. For example:
   `font Courier      /QIBM/UserData/OS400/Fonts/PSFonts/HEL.PFB`
3. Save the new *psfonts.map* file.

PostScript data streams

PostScript data streams are converted by using a separate job named QIMGSERV.
The QIMGSERV job runs in the QSYSWRK subsystem, and uses the QIMGJOBD
job description with a priority of 5. The QIMGSERV job is started by either the
writer job or the job which calls the convert image API. The QIMGSERV job and
the job that starts the QIMGSERV job communicate with each other so that control
is automatic.

How page size is determined

Page size is determined by the output data stream format and the printer device
type. Depending on these values, the page size may be derived from various
places. If the page size is not specified with the convert image API, you can use
the following guidelines to determine from where the page size is derived:

- If the output data stream format is not AFPDS and the printer device is not an
  IPDS printer configured as AFP(*YES), the paper size from Paper Source 1 of the
  device description is used.
- If the output data stream format is not AFPDS and the printer device is an IPDS
  printer configured as AFP(*YES), a default value of letter is used.
- If the output data stream format is AFPDS, the page size is calculated from the
  values in the specified printer file.

Troubleshooting

The following are answers to questions that may arise when you use the image
print transform function or convert image API:

- Why does it take longer to process PostScript data streams?

  One reason why PostScript data streams take a long time to process is the
  amount of information that needs to be transformed. Color documents especially
  require large amounts of memory and many data conversions, which means
  longer processing times.

  Note: If the photometricity of the converted data stream is not requested, it is
        assumed by default to be RGB, or color. However, if you know you do
        not want RGB, or the input data stream is not color, specify an image
        configuration object that only supports black and white output. This will
        greatly increase the throughput of the image print transform function, and
        speed up PostScript processing.

- Why is the converted data stream positioned incorrectly on or off the page? Why
  is it not centered?

  The resolution specified in the image configuration object is probably not
  supported by the printer the object is configured with. When this happens, an
  incorrect no print border is retrieved from the image configuration object and
  the data is consequently positioned incorrectly on the output page. The printer

may also be set up to automatically add a no print border, which will cause the output generated by the image print transform function to be shifted on the page. Verify that the correct image configuration object is being used with the printer, and that the printer has been set up properly and has been physically calibrated.

- Why didn't the PostScript data stream generate a new data stream?

  Chances are that the PostScript data stream did not contain any printable data. To verify this, start the image print transform function and check the job log of the writer. Look for a message that indicates that there was no printable data found. If no message exists, an error may have occurred processing the file. Refer to the job log of the corresponding QIMGSERV job.

- Why is the printed image three times the original size when converted from color or gray scale to black and white?

  When a color image or gray scale image is converted to black and white, a dithering process takes place. In this process, a single color or gray scale pixel is transformed into a 3x3 matrix of pixels. Each pixel within this matrix will be either black or white, depending on the color being rendered.

Additional documentation

For more information, see the following:

- System API Reference in the iSeries Information Center.
- Control Language in the iSeries Information Center.

For additional information on printing and the iSeries server, refer to the following publications:

- IBM AS/400 Printing III (GG24-4028-00)
- IBM AS/400 Printing IV (GG24-4389-00)
- IBM AS/400 Printing V (SG24–2160–01)

Chapter 15. Other printing functions provided by the OS/400 program

The OS/400 program contains the advanced function printing (AFP) support, discussed in Part 3, "Advanced function printing (AFP)" of this guide. It also contains these printing functions:

- PrintManager/400
- Data Description Specifications (DDS)
- Advanced Printer Function (APF)
- Graphical Data Display Manager (GDDM)
- QWP4019 Program

PrintManager/400

PrintManager is the collective name of a group of IBM licensed programs or operating system functions designed to provide common access to printing, including advanced function printing (AFP), across the supported environments. PrintManager/400 provides the PrintManager functions within the OS/400 operating system and IBM SAA® PrintManager provides those functions in the VM and MVS operating systems.

PrintManager/400 consists of the following:

- The OS/400 implementation of the PrintManager Interface, an element of the IBM Systems Application Architecture® (SAA) common programming interface (CPI). This implementation allows you to write portable applications for sending spooled files to an output queue for printing.

  Among other benefits, the PrintManager Interface allows you to specify and validate print-option values from within an application in a form that is consistent across the supported environments. Applications that use the PrintManager Interface, therefore, are portable because they can be developed for one environment and used with little or no modification in another environment.

- An application programming interface (API) that allows you to create print descriptors that can contain common information about printer routing, printer capabilities, and printer and job defaults. Applications that use the PrintManager Interface can use print descriptors created by the API. To create and maintain print descriptors, you write C language applications using the API verbs.

PrintManager/400 provides C programming language support for the API, and C, COBOL, and RPG programming language support for the PrintManager Interface.

PrintManager defines a set of print options that are consistent across the supported environments, and it allows you to specify these print options within the application. PrintManager also provides the ability to:

- Select AFP resources from a system library on the printing system
- Package AFP resources with the print job (inline) when you send the job from one system to another.

These print options, combined with the ability to create common, portable applications and printing definitions (with print descriptors), make possible easy

and consistent access to printing across your organization. With PrintManager/400 you can use the PrintManager Interface and API to create batch applications or installation-specific end-user interfaces for printing.

For more information about the PrintManager Interface, refer to *Systems Application Architecture Common Programming Interface PrintManager Reference* , S544-3698. For more information about the API, refer to *PrintManager Application Programming Interface Reference* S544-3699.

Data Description Specifications (DDS)

The DDS Reference topic in the iSeries Information Center contains detailed instructions for coding the data description specifications (DDS) for printer files that are described externally.

This allows the programmers to add more flexibility in defining how their printed output will look.

Advanced Printer Function

The advanced printer function is intended for system and application programmers to use in designing changed-to-tailor forms.

The advanced printer function utility (APF) allows you to create and maintain changed-to-tailor forms by using special print capabilities available on the 5224 Models 1 and 2 and 5225 Models 1, 2, 3, and 4 dot matrix printers. APF can create the background information needed to make it appear as though you are using a preprinted form or a printer with a variety of special fonts.

Functions of APF

APF provides support to allow you to:
- Design the layout of a form
- Specify fields on a form that will contain special features
- Design special features for a form
- Produce blank copies of a form
- Merge spooled data with a predefined form

The special features you can design with APF include:
- Logos, special symbols, or large characters
- Bar codes [4]
- Bar charts
- Constant fields, such as column headings
- Vertical and horizontal lines, that can be used to form boxes
- Highlighted fields
- Underlined fields

---

4. The bar codes that this utility prints are representations of Code 3 of 9, EAN (8 digit and 13 digit), changed PLESSEY (MSI), and UPC (versions A and E) bar codes. Test all bar codes you print on the 5224 Printer or 5225 Printer to make sure the wanding or scanning devices you use can read the codes created. Nonglossy paper is recommended.

Graphical Data Display Manager (GDDM)

GDDM lets you add color and pictures to application programs by having your application programs pull in user-created GDDM routines.

GDDM routines perform basic graphics tasks, such as drawing a line from point A to point B. Also, GDDM routines are called in an application program to start and stop the graphics environment, set color and width of a line, send the picture to the work station, and so forth.

GDDM works with the following high-level languages (HLL):
- BASIC
- COBOL/400
- RPG/400
- PL/I

Required iSeries Server Hardware

You can write and run GDDM application programs using any model of the iSeries and AS/400 server that has the Operating System/400 program installed.

Although you can write and compile the programs on any work station that has been described to the system, only the following devices can be used to display graphics:
- IBM Personal Computer with work station function
- IBM Personal Computer with work station emulation
- 5292 Model 2
- IBM Personal Computer with 5250 emulation

In this manual the term "graphics work station" means one of these devices.

On the IBM Personal Computer and IBM Personal System/2, the graphics configuration of the device is determined by the hardware capability and the virtual device interface (VDI) driver that is loaded in the CONFIG.SYS file on the personal computer or Personal System/2. The following table describes the capabilities of each VDI driver:

VDI Driver	Resolution	Colors	Gray Levels
VDIDY004	320 x 200	4	
VDIDY006	640 x 200		2
VDIDY00D	320 x 200	8	4
VDIDY00E	640 x 200	8	4
VDIDY00F	640 x 350		4
VDIDY010	640 x 350	4/8	2/4
VDIDYPGD	640 x 480	8	8
VDIDYA11	640 x 480		2
VDIDYA12	320 x 200	8	8
VDIDY011	640 x 480		2
VDIDY012	640 x 480	8	8
VDIDY013	320 x 200	8	8
VDIDYAF1	1024 x 768	8	8
VDIDYAF2	1024 x 768	8	8

These plotters can be attached to graphics work stations:

- IBM 6180 Plotter
- IBM 7371 Plotter
- IBM 7372 Plotter

Graphics can be printed on these SNA character string (SCS) devices:
- IBM 4214 Printer
- IBM 4234-2 Printer
- IBM 5224 Printer
- IBM 5225 Printer

or these intelligent printer data stream (IPDS) devices:
- IBM 3112 Printer with IPDS feature
- IBM 3116 Printer with IPDS feature
- IBM 3812 Printer
- IBM 3816 Printer
- IBM 3912 Printer with IPDS feature
- IBM 3916 Printer with IPDS feature
- IBM 4028 Printer
- IBM 4224 Printer
- IBM 4230 Printer
- IBM 4234-8, or 4234-12 Printer
- IBM 4312 Printer with IPDS feature
- IBM 4317 Printer with IPDS feature
- IBM 4324 Printer with IPDS feature
- IBM 6400 Printer with IPDS feature
- IBM 6408 Printer with IPDS feature
- IBM 6412 Printer with IPDS feature

It is also possible to send a graphics data format (GDF) file (the internal data GDDM interprets to draw the picture) to other systems. The device receiving the graphics data must have the software necessary to interpret the data.

Required OS/400 Software

Besides having the OS/400 program installed, you must have a compiler for one of the following high-level languages:

BASIC
    IBM AS/400 BASIC Version 2, Program 5738-BA1

RPG/400
    IBM SAA AD/Cycle* RPG/400 Version 2, Program 5738-RG1

COBOL/400
    IBM SAA AD/Cycle® COBOL/400 Version 2, Program 5738-CB1

PL/I   IBM AS/400 PL/I Version 2, Program 5738-PL1 (with library QGDDM in your library list)

Required Knowledge

To write graphics application programs for the iSeries server, you must know OS/400 application programming in one of the four high-level languages.

QWP4019 Program

QWP4019 is an IBM-supplied program that you can call to set flags on and off in a printer device description. Setting the flags on enables functions not accessible through the Create Device Description (Printer) (CRTDEVPRT) or Change Device Description (Printer) (CHGDEVPRT) commands.

The QWP4019 program allows you to take advantage of functions available on an attached printer but not supported by the emulator you are using.

Note: The QWP4019 program was designed to make the 4019 fonts available for the IBM LaserPrinter 4019 printer using emulation. Additional parameters can be specified to enable functions in the SCS printers. Most of these parameters are only valid for printers considered as a 5219 or 3812.

QWP4019 Parameter Names and Functions

The following list contains the QWP4019 parameter names and explains the function provided when they are called.

Parameter
> **Function Provided**

*ON   This parameter sets a flag in the printer device description that:
> - Indicates to OS/400 that the 4019 fonts are to be used instead of the 5219 or 3812 fonts. See Table 37 on page 404 to view the font mapping and substitution for the 4019 printer.
> - Enables manual paper feed selection if the value in the form feed (FORMFEED) parameter is *CUT.
> - Prevents the value of *COR in the page rotation (PAGRTT) parameter from being sent to a 3477 InfoWindow display that has an attached printer configured as a 5219. This is important because the 3477 does not support computer output reduction (COR). Without this flag, the PAGRTT(*COR) value in the printer file cannot be used for these printers.

*OFF   This parameter sets the *ON flag off. Also, if the *SIC or *COR flags were set on, using the *OFF parameter sets them off.

*CHECK
> This parameter asks how the printer appears to the iSeries server (as a 3812, 5219, or 4019).
>
> If the value 4019 is returned, it indicates that the QWP4019 program has been called with the *ON parameter.

*CNT   This parameter sets a flag in the printer device description that tells the iSeries server that the printer has a continuous form feed device. The flag is used by the system to determine whether forms alignment is possible for printers that are configured as a 3812 printer. The 3812 printer does not support continuous forms.
>
> This flag can only be set for printers that are configured as a 3812 printer.

*CNTOFF
> This parameter sets the *CNT flag off.

*IMP   This parameter sets a flag in the printer device description that allows a print quality selection other than draft when:

- The value in the page rotation (PAGRTT) parameter of the printer file is *AUTO.
- The printer is attached to an emulator (for example, a 3477 InfoWindow display) that supports page rotation.

For automatic page rotation to be done by an actual 3812 SCS printer, the print quality control sent to the printer must specify draft quality. Therefore, without the *IMP flag on, the iSeries server sends controls to the printer for draft selection. Because the 3812 SCS printer supports only one level of print quality, printed output is not affected by print quality selection.

With the *IMP flag set on, the iSeries server sends the value in the print quality (PRTQLTY) parameter of the printer file directly to the emulator. This is done instead of changing the print quality to draft when the value of the page rotation (PAGRTT) parameter is *AUTO.

*IMPOFF
This parameter sets the *IMP flag off.

*SIC This parameter sets a flag in the printer device description that sends an ASCII Set Initial Conditions command to the printer. This command turns off the font intervention messages for a 4019 or 4029 printer.

The 4019 or 4029 printer must be attached to a 3477 InfoWindow display and be configured as a 5219 printer.

The *ON flag must be set in the 4019 or 4029 device description.

Note: Do not attempt to turn this flag on for any device other than a 4019 or 4029 attached to a 3477 InfoWindow display.

*SICOFF
This parameter sets the *SIC flag off.

*COR This parameter sets a flag in the printer device description that enables computer output reduction (value of *COR in the page rotation (PAGRTT) parameter of the printer file.)

*COR is only needed if the *ON flag is set.

The printers must be attached to a 348x InfoWindow, configured as a 5219, and have the *ON flag set in the printer device description.

To the iSeries server, printers configured as 5219 appear the same whether they are attached to a 3477 or to a 348x InfoWindow display.

The *ON flag prevents computer output reduction from being done on printers attached to the 348x or 3477. This is because the 3477 does not support computer output reduction. Therefore, if the *ON flag is set, the *COR flag must be set on to enable computer output reduction for printers configured as 5219 and attached to a 348x InfoWindow display.

*COROFF
This parameter sets the *COR flag off.

*RST This parameter sets a flag in the printer device description that causes the writer to reset the printer at the beginning of each spooled file. In most environments, this may cause a noticeable decrease in performance because of the SNA overhead involved in doing the reset. This flag can be set for any SCS printer.

*RSTOFF

This parameter sets the *RST flag off.

*ON5256

This parameter causes the operating system to configure the printer as a 5256 printer

*ON5262

This parameter causes the operating system to configure the printer as a 5262 printer

*OFF52

This parameter turns off the *ON5256 and *ON5262 flags in the device description.

*ON4214

This parameter causes the operating system to configure the printer as a 4214 printer.

*OF4214

This parameter turns off the *ON4214 flag in the device description.

How Does the QWP4019 Program Work?

The QWP4019 program sets flags in the printer device description through calls to the program with certain parameters specified. For example,

```
CALL QWP4019 (PRT01 *CNT)
```

sets *CNT on and tells the iSeries server that PRT01 has a continuous form feed device.

Flags are set off by calling QWP4019 with certain parameters specified. For example,

```
CALL QWP4019 (PRT01 *OFF)
```

sets the 4019 flag off in the device description for PRT01.

Since the flags are stored in the device description, QWP4019 only needs to be run once for each printer and each function. The flags can be changed only by running QWP4019 or deleting the device description. To make sure the flag is set, it is recommended that the printer writer for that printer be restarted after the QWP4019 program has been called.

QWP4019 Program Examples

Following are examples that show how to use the QWP4019 program.

Example 1

Setting the 4019 flag on and then off in the device description for PRT01.

QWP4019 CALL	Result
CALL QWP4019 (PRT01 *ON)	Sets the 4019 flag on in the device description for PRT01.
CALL QWP4019 (PRT01 *CHECK)	The system returns 4019 because the 4019 flag is set on.
CALL QWP4019 (PRT01 *OFF)	Sets the 4019 flag off in the device description for PRT01. **Note:** This CALL also sets flags *SIC and *COR off.

QWP4019 CALL	Result
CALL QWP4019 (PRT01 *CHECK)	The system returns 5219 or 3812 because the 4019 flag is off.

Example 2

Enabling COR and the 4019 fonts for a 4019 printer attached to a 348x InfoWindow display.

QWP4019 CALL	Result
CALL QWP4019 (PRT01 *ON)	Sets the 4019 flag on in the device description for PRT01. This supplies the 4019 fonts but disables computer output reduction.
CALL QWP4019 (PRT01 *COR)	Sets the *COR flag on in the device description. This enables computer output reduction for PRT01.

Example 3

Enabling near letter quality (NLQ) print quality for a IBM Personal Printer Series II 2390 printer attached to a 3477 InfoWindow display.

QWP4019 CALL	Result
CALL QWP4019 (PRT01 *IMP)	Turns the *IMP flag on in the device description for PRT01.
CALL QWP4019 (PRT01 *CHECK)	The system returns 5219 or 3812 because the 4019 flag is off. There is no check for *IMP, *SIC, or *COR flags.

The following table provides a summary of the QWP4019 parameters, the call used to set flags on, and the call used to set flags off.

Table 30. QWP4019 Parameter Names and Calls

Parameter Name	Call to Set Flag On	Call to Set Flag Off
*ON	CALL QWP4019 (PRT01 *ON)	CALL QWP4019 (PRT01 *OFF)[1]
*IMP	CALL QWP4019 (PRT01 *IMP)	CALL QWP4019 (PRT01 *IMPOFF)
*SIC	CALL QWP4019 (PRT01 *SIC)	CALL QWP4019 (PRT01 *SICOFF)
*COR	CALL QWP4019 (PRT01 *COR)	CALL QWP4019 (PRT01 *COROFF)
*CNT	CALL QWP4019 (PRT01 *CNT)	CALL QWP4019 (PRT01 *CNTOFF)
*RST	CALL QWP4019 (PRT01 *RST)	CALL QWP4019 (PRT01 *RSTOFF)
*ON5256	CALL QWP4019 (PRT01 *ON5256)	CALL QWP4019 (PRT01 *OFF52)
*ON5262	CALL QWP4019 (PRT01 *ON5262)	CALL QWP4019 (PRT01 *OFF52)
*ON4214	CALL QWP4019 (PRT01 *ON4214)	CALL QWP4019 (PRT01 *OF4214)
Note:		
[1] Using the CALL QWP4019 (PRT01 *OFF) command sets the *SIC, *COR, and *ON flags off.		

Chapter 16. Other printing functions provided by licensed programs and iSeries server hardware

Printing functions are provided by the following IBM licensed programs:
- Advanced Function Printing Utilities/400
- Business Graphics Utility (BGU)
- iSeries Access for Windows
- TCP/IP Connectivity Utilities/400

A **licensed program** is a separately orderable program, supplied by IBM, that performs functions related to processing user data.

Printing functions are provided by these IBM products:
- IBM InfoWindow* 3477
- ASCII Work Station Controller

Advanced Function Printing Utilities/400

What is AFP Utilities/400?

The IBM Advanced Function Printing Utilities/400 is a licensed program (Program 5722-AF1). It provides utilities that work together to reduce the cost of printing text and images on paper when using advanced function printing (AFP) on the iSeries server.

The AFP Utilities make the most efficient use of an IPDS-capable printer, provide a display screen editor that allows you to see the actual position-by-position design of the overlay, and are supported on all displays that attach to the iSeries server. The following make up the AFP Utilities:
- Overlay utility
- Print format utility
- Resource management utility

The information in this guide identifies and describes the major elements of the AFP Utilities.

Overlay utility

The overlay utility allows you to design electronic form overlays.

You can use overlays to reduce or eliminate the use of preprinted paper forms.

The overlay can be composed of text, image (page segments), lines, boxes, and bar codes. All of the environmental data (font references, location on the page, and so on) is defined as a part of the overlay definition.

The following diagram is an example of an overlay you can create using the AFP Utilities.

Electronically Stored Form Overlay

Variable Page Data

Merge

Finished Product

RV2H309-0

Element descriptions

The overlay created by the AFP utilities is composed of any combination of the following elements:

- Text

  Text can be placed at a specified position on the overlay. The text attribute may also be specified to describe the text characteristics such as font selection, format and highlighting of underline, overstrike, color, degree of rotation, and character enlargement.

  Note: If underlining or overstriking are used in an overlay generated by AFP Utilities/400, that overlay will not print on a System/390 system. The Print Services Facility (PSF) support on the System/390 does not recognize the commands in the overlay which generate the underline and overstrike functions.

- Line

  The line is any straight line vertically or horizontally connecting two points. You may select the type of line (such as dotted, dashed, or solid) and the width of the line.

- Box

  The box can be placed on the overlay by specifying two diagonally opposite corners. You can select the shade pattern inside the box, the type of box line (dotted, dashed, or solid), and the width of the line. Also, you may define the text inside the box. The text may be justified inside the box.

- Bar Code

  A bar code can be placed at a specified position on the overlay. You can specify the following bar code data: position to be placed on overlay, type of bar code, size, color, and whether or not to print Human Readable Interpretation (HRI).

- Page Segment

  You can include a page segment that is in the AFP resource library. You can refer to this page segment by its name and specify print position to define it as an overlay element. You can include the same page segment repeatedly in an overlay. A page segment contains an image, such as a logo, a picture, or a map.

- Graphics

  A graphic can be placed at a specified position on the overlay.

Print format utility

The print format utility allows you to print data from database file members, which are created by application programs or utilities, in various forms such as text or bar codes, and in various formats on IPDS printers.

With the AFP Utilities, the forms and formats for printing are designed by defining a **printout format definition** instead of writing your own application programs.

In addition to the data from database file members, you can print fixed data, such as a logo or a title.

The following diagram is an example of what you can print using the print format utility.

List of Products

Product Name	Price
Color TV	300
AM Radio	20
Video	500
⋮	⋮
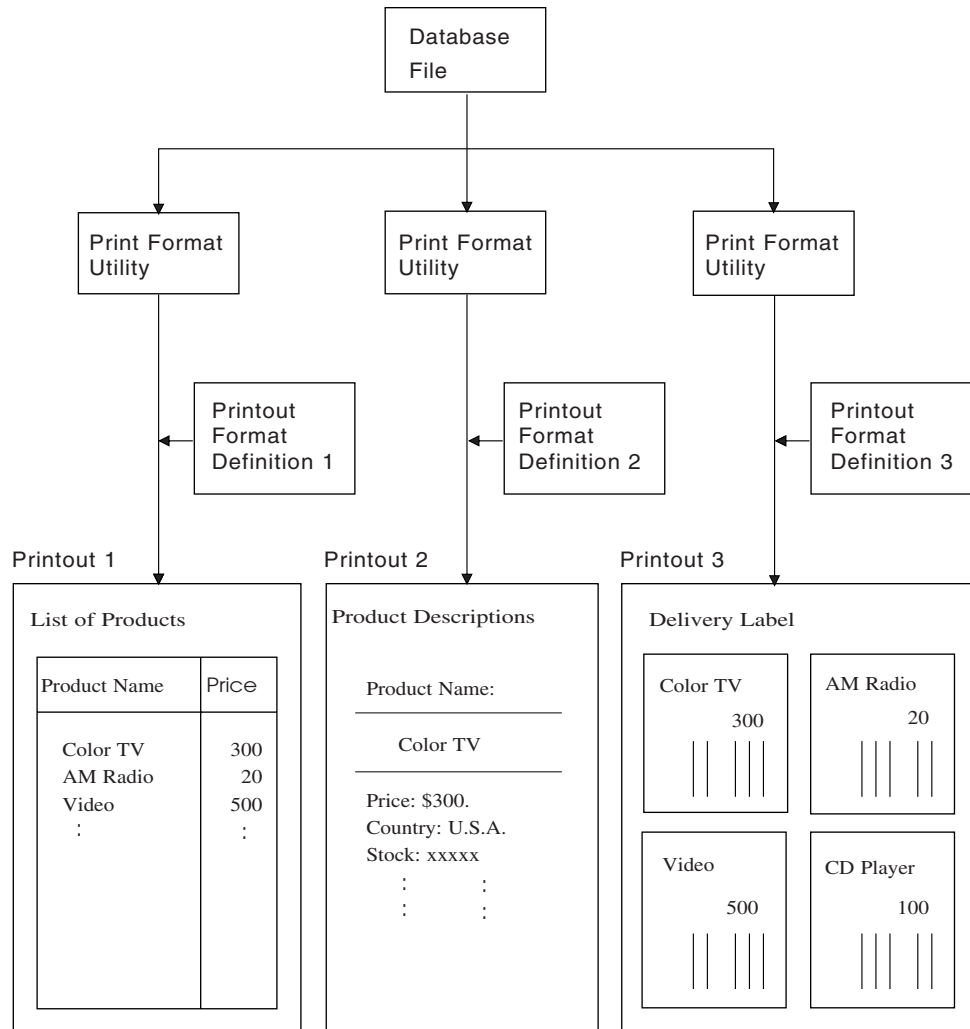
RV2H308-2

Creating the printout format definition

The following list identifies the things you can specify when you create a printout format definition using the Print Format Utility.

- Printout format definition specifications
- Printout format definition fonts
- Name of database file, library, and record format
- Record layout elements (text, images, lines, boxes, or bar codes)
- Page layout elements (record layout, text, images, lines, boxes, or bar codes)
- Record selection
- Printout specifications

Resource management utility

The resource management utility allows you to manage AFP resources such as overlay objects or page segment objects.

The resource management utility has the following functions:

- Creating page segments

- Working with overlays
- Working with page segments

Creating page segments

This function allows you to convert or transform a page segment from an OS/400 database file member or a PC document that contains image data stream data. If the page segment is from a PC document, it can be stored in an OS/400 folder using the iSeries Access for Windows shared folders function.

Working with overlays

This function allows you to:

- Copy an overlay object in a library.
- Delete an overlay object in a library.
- Print an overlay in a library on an IPDS printer. You can also specify that your overlay be printed over the top of a grid. The appearance of the grid under the overlay provides you with an excellent tool for detailed design of an overlay.
- Display the description of an overlay object.
- Put an overlay into a file.
- Change the description of an overlay object.

Working with page segments

This function allows you to:

- Copy a page segment object in a library
- Delete a page segment object in a library
- Print a page segment object in a library on an IPDS printer
- Display the description of a page segment
- Put a page segment into a file.
- Change the description of a page segment

Advanced DBCS printer support/400

The IBM Advanced DBCS Printer Support/400 program (5722-AP1) is designed for users who have DBCS printers. This program provides DBCS support with five utilities:

- Advanced printer writer (APW)
- Advanced page printer writer (APPW)
- Kanji printer function (KPF)
- Print form description and symbol migration aid
- Printer function control (PFC)

The advanced printer writer utility supports printers that are capable of printing Japanese, Korean, Traditional Chinese, and Simplified Chinese including the following:

- Work station printers 4028, 4216, 5317, 5553, 5557, 5563, 5575, 5577, and 5587
- Floor-standing printers 5227 (Models 1, 2, 3, and 5) and 5327 (Models 1, 2, and 3)

The advanced page printer writer utility supports the Japanese 5337 page printer. The Kanji printer function utility supports the Japanese 5583 page printer.

The print form description/symbol migration aid utility converts the form descriptions and symbol files of the System/36 advanced printer writer and Kanji printer function to equivalent form descriptions and symbol files for the iSeries server.

The printer function control utility gives users the capability of printing user data with user-defined single-byte character set (SBCS) fonts.

The following manuals contain more detailed information about Advanced DBCS Printer Support/400:
- *ILE C/C++ for AS/400 MI Library Reference*.
- *Advanced Print Writer*.
- *IBM 5583 Kanji Print Function User's Guide*, N:SH18-2179.
- *System/36 APW/KPF Migration Utilities User's Guide*, N:SH18-2234.

Business graphics utility (BGU)

What is BGU?

The IBM Application System/400® Business Graphics Utility (BGU) is a licensed program (Program 5738-DS1) that allows you to create, change, store, display, print, and plot charts representing data. You can also store, retrieve, delete, change, rename, and copy charts using the *Manage existing chart formats* option. All OS/400 graphics hardware devices are supported.

The following lists the functions and features of BGU.
- Menu-driven interface to powerful business graphics functions.
- Extensive options for creating and changing chart formats.

  Use BGU to create new charts or change existing charts. When you create a chart, you see a series of menus where you can specify the appearance of the chart, including its type, headings, axis definition, margins, legend position and attributes, line types, fill patterns, and colors. Once the chart format is defined, data can be supplied. Because data is handled separately, any suitable data can be displayed or printed with a chart format.
- Varied chart types, including:
  - Bar charts
  - Line graphs
  - Surface charts
  - Histograms
  - Pie charts
  - Venn diagrams
  - Text charts
- Support for missing values.
- Convenient storage, retrieval, copying, renaming, deleting, and changing functions.
- Ability to display charts from the control language (CL) application programs using the Display Chart (DSPCHT) command.
- Ability to save charts in the form of a graphics data file (GDF).
- Option to display, print, or plot a GDF with BGU.
- Charts saved in a GDF can be displayed, printed, or plotted using the Display Graphics Data File (DSPGDF) command.

- Access to IBM graphics hardware for iSeries servers:
  - Display terminals
    - IBM Personal System/2 (PS/2) Models 50, 60, and 80 with IBM work station emulation
    - IBM Personal Computer or PS/2 Models 25 and 30 with Enhanced 5250 Display Station Emulation
    - IBM Personal Computer or PS/2 (all models) with work station function
    - IBM 5292 Model 2 Color Display Station
  - Plotters
    - IBM 7372 six-pen Plotter
    - IBM 7371 two-pen Plotter
    - IBM 6180 Plotter configured as an IBM 7372 or 7371
  - Printers
    - IBM 3112 Printer with IPDS feature
    - IBM 3116 Printer with IPDS feature
    - IBM 3812 IPDS Printer
    - IBM 3816 IPDS Printer
    - IBM 3912 Printer with IPDS feature
    - IBM 3916 Printer with IPDS feature
    - IBM 4028 Printer
    - IBM 4214 Printer
    - IBM 4224 IPDS Color Printer
    - IBM 4230 Printer
    - IBM 4234 Model 2 Printer
    - IBM 4234 Model 12 Printer
    - IBM 4312 Printer with IPDS feature
    - IBM 4317 Printer with IPDS feature
    - IBM 4324 Printer with IPDS feature
    - IBM 5152 Model 2 Printer
    - IBM 5224 Printer
    - IBM 6400 Printer with IPDS feature
    - IBM 6408 Printer with IPDS feature
    - IBM 6412 Printer with IPDS feature
    - IBM 5225 Printer

Data access capability

Data values can be typed or taken from a database file. Charts displayed from typed data allow for graphing and manipulation of data. Charts using database files for data input offer on-demand charts to display the most current user data.

Data retrieved from database files can also be changed from a keyboard, allowing the user to manipulate data from analysis, such as "what if" conditions.

iSeries Access for Windows

The iSeries server is well suited to support personal computers and personal computer printers (hereafter called personal printers). In your daily operations you may have need for data that is created and stored on your personal computers to be printed on system printers. Conversely, you may have requirements for data created and stored on your iSeries server to be printed on personal printers.

Network printer function

The network printer function is one of the functions available with the licensed program iSeries Access for Windows, which runs on both your iSeries server and your personal computer.

You use the network printer function to print data from your PC application programs on iSeries server printers. The **network printer function** allows you to use a printer attached to the host system as though the printer were directly attached to a personal computer.

The PC data could be as simple as a personal computer print screen, or as complicated as a word processing document.

There are several reasons why you might want to use the network printer function. For example, you can take advantage of the faster speed or better quality of the large iSeries server printers. Additionally, if your personal computer work station does not have a personal printer attached, you could print your PC data on iSeries server printers.

The diagram below provides an example of the network printer function concept.

Sending Output

AS/400 System

AS/400 3812 Printer

Printed Output

RSLN425-1
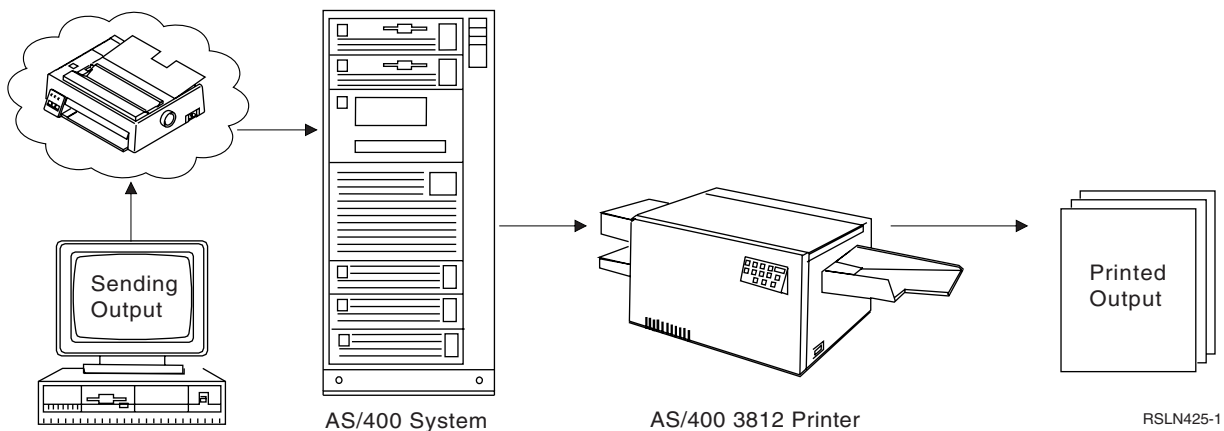
For this example, you will need to set up your PC application programs as if they were sending output to an IBM Proprinter, Model 4201. The network printer function intercepts and routes the PC output to an OS/400 spooled file. Since iSeries server printers like the 3812 are not designed to print data from personal computers, the network printer function converts the PC data to SNA character string (SCS) data.

Personal computer operating systems

The network printer function can run on the DOS, OS/2, or Windows operating systems.

Windows and advanced function printing (AFP) using the network printer function

The iSeries Access for Windows network printer function provides an AFP printer driver for users of Microsoft Windows 3.0 or later and one for users of OS/2 2.0 or later. Using either of these printer drivers, OS/400 users can print their application output on AFP-configured IPDS printers.

The printer driver converts the standard print format generated by Windows or OS/2 applications to an all-points-addressable raster format, suitable for printing on AFP-configured IPDS printers.

The AFP-capable IPDS printers must be configured DEVTYPE(*IPDS) and AFP(YES). This is done in the printers device description. The format generated by the printer drivers will not drive the IPDS printers at rated speed. Examples of IPDS printers that support AFP are: 3812, 3816, 3820, 3825, 3827, 3828, 3829, 3835, and 4028.

The printer drivers also have an option to convert the Windows or OS/2 print format into AFP page segment or overlay resource objects. These AFP resource objects can then be moved to the iSeries server using the iSeries Access for Windows transfer function, to be used by applications printing on AFP-configured IPDS printers.

Printer emulation

Printer emulation is the printing of data from a host (iSeries server, S/36) to personal printers connected to personal computers (PC). Host applications generate print jobs that can be printed on host attached printers, or printers attached to PCs. Where they actually print is transparent to the host application. iSeries Access for Windows is a recommended Emulator; Personal Communications and Rumba are popular, also. There is a wide and growing list of PC printers with laser printers being quite popular. IBM4019, IBM4039, HPLJ_IV, and various DESKJET printers are examples.

Print emulation is the receiving of data in HOST format or languages and converting it to PRINTER format and languages. PC host printer languages include PCL, Postscript, and IBM PPDS. iSeries Access for Windows emulates the SCS language, converting that to any printer language. A wide range of formatting is available.

AS/400 System   3812 Printer   Receiving Output   PC   4019 Printer   Printed Output   RV2H321-1
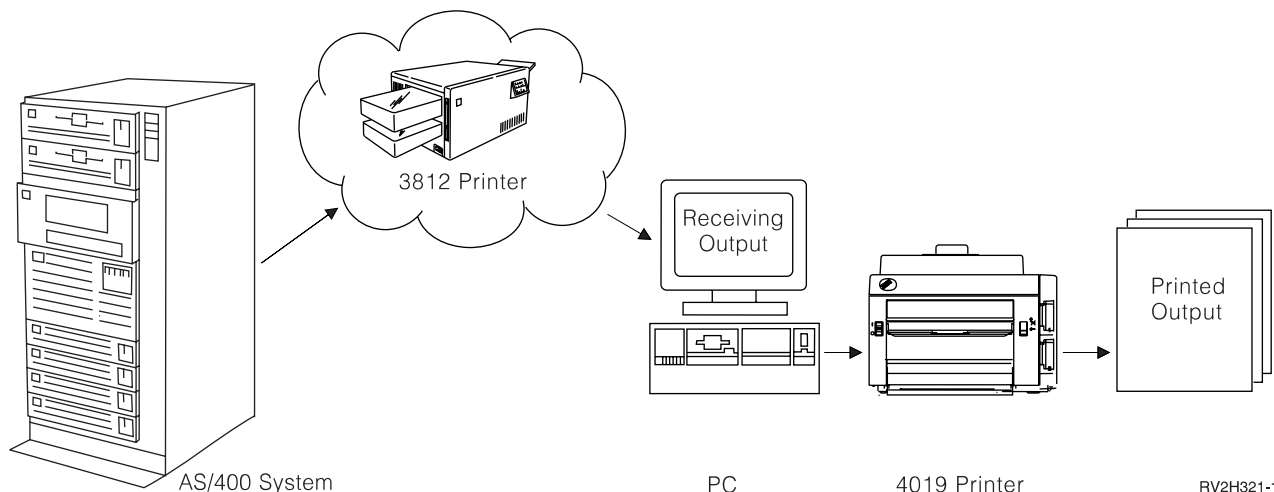
Personal computer operating systems

Printer emulation can run on the DOS, OS/2, or Windows operating systems.

To assist you in working with printer emulation using **OS/2**, the following products are available:

- Communications Manager/2
- RUMBA/400

Contact your IBM representative for more information on these products. If you already have these products, consult the documentation that accompanies them.

OS/400 host print transform function

The host print transform function function can also be used to print data on personal printers. When this function is enabled, SCS data or AFP is transformed to ASCII data and passed through emulators to the specified ASCII printer. If the data stream is ASCII, the data is sent directly to the printer, and not converted by one of the emulators. For more details on using the host print transform function, see Chapter 13, "Working with the host print transform function" on page 271.

Introducing sharing personal printers

You can share a personal printer among several personal computer work stations. Sharing personal printers is a special case of the network printer function used in conjunction with printer emulation.

The personal computers are attached to the iSeries server using either iSeries Access for Windows, OS/2 with Communications Manager/2, or RUMBA/400. Each personal computer work station that shares the personal printer uses the network printer function of iSeries Access for Windows.

The diagram below shows how you can share personal printers using the specific example of attaching the IBM 4019 LaserPrinter to your iSeries server as a iSeries Access for Windows work station performing printer emulation.

| | LPT2 | LPT2 | |
| AS/400 | Personal Computer | Personal Computer | Personal Computer | IBM 4019 Laser Printer (emulated printer) |

RV3H049-1
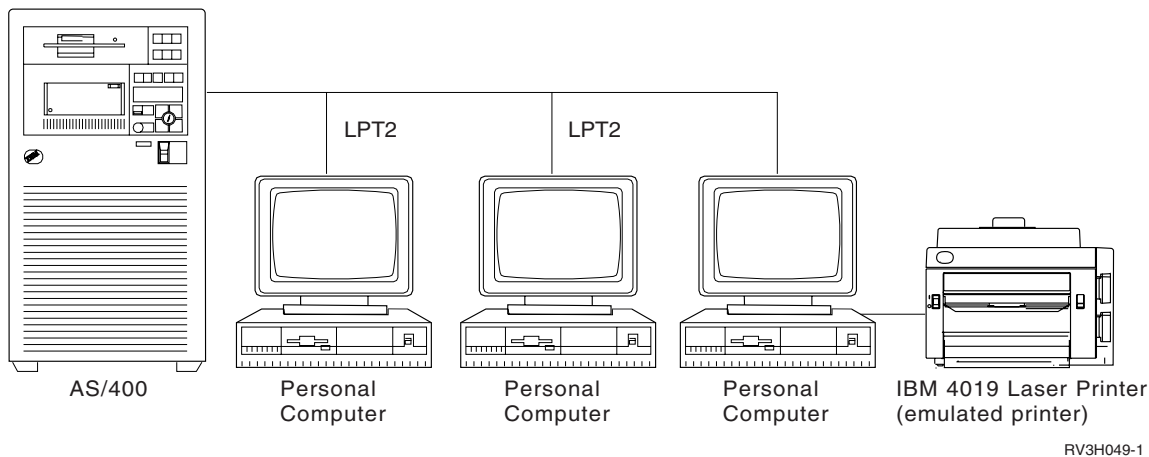
Your PC application programs are set up as if they are sending output to the IBM 4019 LaserPrinter that you are sharing. The network printer function is set up with the ASCII data type. The network printer function intercepts and routes the output

to an OS/400 spooled file. Since the PC data is going to print on a personal printer, the IBM 4019 LaserPrinter, the network printer function does not convert the PC data.

IBM InfoWindow 3477, 3486, 3487, and 3488 printer support

The IBM InfoWindow* display station can be locally attached to the iSeries server or remotely attached to the IBM 5294 or 5394 Remote Control Unit via twinaxial cable. The InfoWindow has a printer port that can support attachment of most IBM personal printers. Examples of such personal printers are:
- IBM LaserPrinter, Model 4019
- IBM Quietwriter® III, Model 5202
- IBM Quickwriter, Model 5204
- IBM Proprinter II, Model 4201

The attached printer can be used either as a local screen printer or as a system printer to print OS/400 spooled jobs (for example, printing an OfficeVision document or a job generated on a personal computer using the network printer function).

There are several advantages to using personal printers that attach to the InfoWindow. The lower cost and smaller size of personal printers makes it convenient to put personal printers at your display stations that are attached to your iSeries server.

Note: If you specify `local` for the type style option for printer setup on a 3477, 3486, 3487, or 3488 InfoWindow display, font selection or substitution by the printer may give you unpredictable results.

For further information about attachment of personal printers to the 3477, see the following manuals:
- *IBM InfoWindow 3477 User's Guide* , GA18-2923
- *Connecting Personal Printers to IBM Systems* , S544-4209

ASCII work station controller

The ASCII Work Station Controller provides the ability to attach ASCII displays, ASCII printers, and personal computers to the iSeries server through the RS232 or RS422 interface. The work station controller supports attachment of a wide range of printers. Examples of such personal printers are:
- IBM Proprinter II, Model 4201
- IBM LaserPrinter, Model 4019
- IBM Quickwriter, Model 5204
- IBM 4234, Model 13

The work station controller also provides the ability to attach a personal computer running iSeries Access for Windows work station function. A personal printer attached to the personal computer can be used as an iSeries server printer

For further information about attachment of printers to the work station controller, see the following manuals:
- *ASCII Work Station Reference*
- *Connecting Personal Printers to IBM Systems* , S544-4209

Sending and printing files with TCP/IP

The TCP/IP Connectivity Utilities for OS/400 is a licensed program (5722-TC1) that provides connection services to other systems. Important to printing is the spooling and print support available with this licensed program.

You can request to have your spooled files sent and printed on any system in your TCP/IP network. The term often used by UNIX** TCP/IP software to describe this support is **line printer requester (LPR)**. The LPR is the sending, or client portion, of a spooled file transfer. On the iSeries server, the Send TCP/IP Spooled File (SNDTCPSPLF) command provides this function by allowing you to specify what system you want the spooled file printed on and how you want it printed. There is also a TCP/IP command, LPR, that provides the same parameters and function as the SNDTCPSPLF command. The LPR can also transform an SCS or AFPDS spooled file to ASCII by using the host print transform function before sending the file.

The printing of the file is done by the printing facilities of the destination system. The destination system must be running TCP/IP. On the iSeries server, the **line printer daemon (LPD)** is the process on the destination system that receives the file sent by the SNDTCPSPLF command. The LPD process places the spooled file on a local printer queue. To print the spooled file, the spooled file must be put on an output queue already started to an active printer writer or a writer must be started to that output queue.

For additional information about printing using TCP/IP, see the following manuals:
* TCP/IP Configuration and Reference
* *IBM AS/400 Printing III, GG24-4028*

Part 5. Network printing

Chapter 17. Network Printing

This chapter covers some of the printing functions available on the iSeries server when it is part of a communications network.

The examples in this part of the guide are intended to show you different methods you can use to send data, within a network, to be printed on an iSeries server.
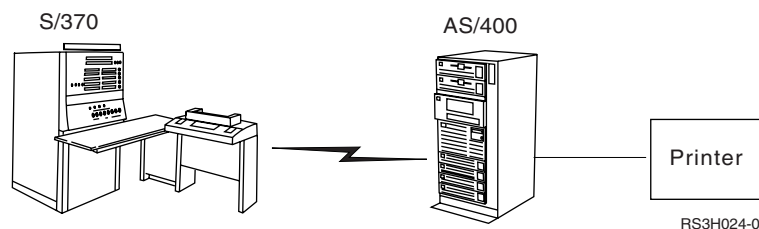
Figure 7. Communications Protocols Used by RJE

3270 Printer Emulation

When configuring 3270 printer emulation on OS/400, you must decide which communications line protocol to use. The iSeries server supports both binary synchronous communication (BSC) and Systems Network Architecture (SNA) 3270 printer emulation. Therefore, the choice is normally dictated by the protocol supported by the System/390 host communications software and controller.

BSC 3270 Printer Emulation

BSC 3270 printer emulation can be used with any System/390 host system that supports a 3274 control unit (Model 51C) in a BSC multipoint tributary network using a nonswitched line. BSC 3270 printer emulation supports the following features:

- Emulation on any line attached to an iSeries server.
- Ability to print information received from the host system on any printer normally supported by the iSeries server.
- Emulation of 3284, 3286, 3287, 3288, or 3289 host printers.

Configuring for BSC 3270 Printer Emulation

Details for configuring for BSC 3270 printer emulation are in Chapter 2 in the 3270 Device Emulation Support manual.

SNA 3270 Printer Emulation

SNA 3270 printer emulation enables the iSeries server to connect into any existing SNA 3270 network. Any printers attached to the system print information received from the System/390 host. The 3270 emulation session is linked to a printer file, which in turn is linked to a printer device. All 3270 printers are emulated as LU 3 printers with the exception of the 3287 and 3289 printers, which may be emulated as either LU 1 or LU 3 printers.

The difference between LU 1 and LU 3 printers is essentially the level of intelligence which the System/390 host expects to find at the printer device. If the printer is emulated as an LU 1 printer, an LU-LU session can be established with

the host. Depending on the host applications, such a printer would be able to fully support SNA character string (SCS) data streams sent to it. If the iSeries server printer emulates an LU 3 printer, only an LU-PU session can be established between it and the System/390 system. This could cause unexpected results when using host print applications. For example, if the host application was expecting to communicate with an LU 1 printer, the printed output would probably not be correctly formatted in terms of line spacing, page length, and so on.

For detailed information on printer emulation, see the 3270 Device Emulation Support manual.

RJE Printing

Remote job entry (RJE) allows an OS/400 user to use System/390-based applications and data in combination with OS/400 functions. RJE is part of the IBM Communications Utilities licensed program (5722-CM1), which runs on all iSeries servers with communications capabilities installed.

When using RJE, the iSeries server functions as a remote work station that submits jobs to a System/390 host for processing under one or more of the following systems:
- MVS/SP™ JES2 (SNA and BSC)
- MVS/SP JES3 (SNA and BSC)
- VM/SP RSCS (BSC only)
- DOS/VSE POWER/VSE (SNA only)
- OS/VS1 RES (SNA and BSC)

RJE can communicate with the host system using either of the following line protocols:
- Systems Network Architecture (SNA) over point-to-point switched or nonswitched connections
- Binary synchronous communications (BSC) over point-to-point switched or nonswitched connections

For BSC, multiple user-defined RJE subsystems may be used concurrently on different lines, each subsystem operating independently of the other. For SNA, multiple subsystems can share the same line.

A writer (printer or punch) output stream can be directed to one of the following:
- A printer file (spooled or nonspooled)
- A physical database file member
- A diskette file (spooled)
- A DDM file
- User program

Configuring for RJE Printing

See the Remote Job Entry (RJE) Guide, chapters 2, 3, and 4, for detailed information on installing and configuring RJE. This task requires a knowledge of data communications.

Communications Line Protocols for RJE

A communications line physically connects an iSeries server to the host. It can be a remote link operating over a long distance, or a cable connecting them on a local area network. The protocol used depends on whether you are in an SNA or a BSC environment, as shown in the following graphic.
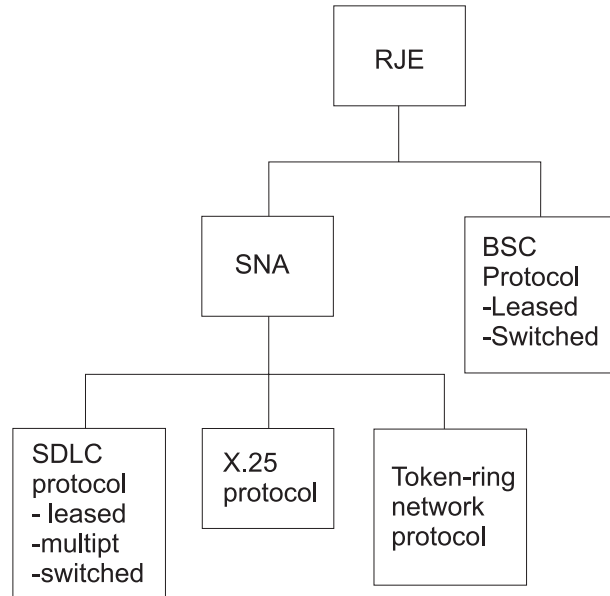
RBAFT527-0

The physical interfaces supported by RJE are shown in Table 31.

Table 31. Physical Interfaces Supported by RJE

	SDLC	X.25	Token-Ring Network	BSC
V.35	Yes	No	No	Yes
X.21bisV.35	Yes	Yes	No	Yes
RS232/V.24	Yes	Yes	No	Yes
X.21bisV.24	Yes	Yes	No	Yes
X.21	Yes	Yes	No	No
Medium Access Control (MAC) IEEE 802.5 / ISO 8802-5	No	No	Yes	No

Printer Files

A destination must be defined for each writer specified in the RJE session description. If a printer file is used, it must have a description that the iSeries server can use to transfer data to a device. The Create Remote Job Entry Configuration (CRTRJECFG) command uses the following command to create a printer file:

```
CRTPRTF  FILE(&CFGLIB/QxxPRTFPRn); +
         CTLCHAR(*FCFC) +
         OUTQ(&OUTQ); +
         MAXRCDS(*NOMAX) +
         SCHEDULE(*FILEEND) +
         LVLCHK(*NO) +
         TEXT('RJE output file for Printer PRnn').
```

The destination for the writer data may be a device file (printer or diskette file), a physical database file, a DDM file, or a user program.

Notes:

1. RJE printer output streams received from the host system can be directed to a system printer.

2. If these RJE printer files are used by any high-level language program, they have to be *program-described.* The term program-described means that the file was created without DDS source describing record formats and fields. Externally described printer files (with DDS source) are not supported under these conditions.

RJE Writer Function

The RJE writer function receives output streams from the host system and sends them to a printer, database file, diskette, or user program.

RV2H332-0
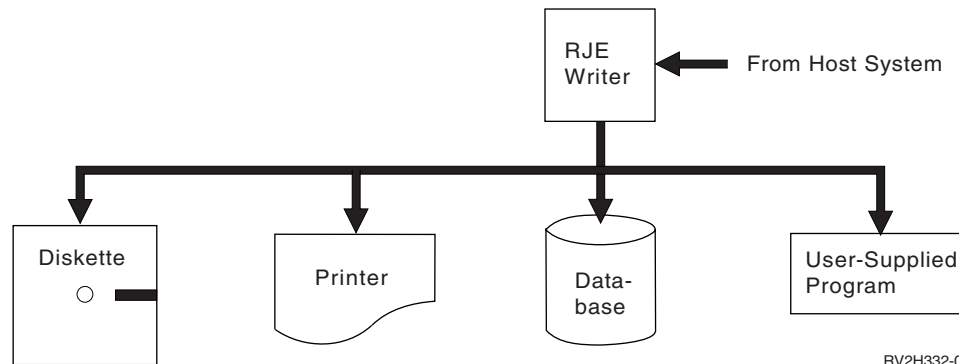
RJE supports up to seven independent printers for BSC, and up to fifteen independent printers for SNA. The actual number supported depends on the host. Printer output streams consist of printed data in extended binary-coded decimal interchange code (EBCDIC) character format. The output streams can be routed by:

*   A writer entry in the session description
*   Parameters specified in the Start RJE Writer (STRRJEWTR) command
*   A forms entry in the FCT.

RJE Forms Control Table

The RJE forms control table (FCT) assigns special processing requirements for specific output streams received from the host system. The FCT functions as a routing table for sending data to a specific output file.

The FCT also functions as a translation table by converting the forms name or number used by the host system into a corresponding name or number more meaningful to the OS/400 user.

When the host system sends a peripheral data set information record (PDIR) (SNA), or a forms-mount message (BSC), the FCT is searched for the host-system-defined name or number. If the name or number is found, the corresponding OS/400 form name for that entry is sent in the forms mount message to the system operator. Both names must be placed in an FCT entry. For example:

```
ADDFCTE  FCT(MYRJELIB/MYRJEFCT) +
         FORMTYPE(X2QH) +
         LCLFORM(INVOICE)
```

When a forms-mount message containing the name X2QH is received from the host system, the name INVOICE is used in the forms-mount message to the system operator. There is no limit to the number of FCTs that can be defined, but each RJE session uses only one at a time.

Controlling the FCT: The following commands are supplied with RJE:

CRTFCT
>   Create Forms Control Table

CHGFCT
>   Change Forms Control Table

DLTFCT
>   Delete Forms Control Table

WRKFCT
>   Work with Forms Control Table

ADDFCTE
>   Add Forms Control Table Entry

CHGFCTE
>   Change Forms Control Table Entry

RMVFCTE
>   Remove Forms Control Table Entry

The following is an example of FCT creation:

```
CRTFCT  FCT(MYRJEFCT)
        AUT(*ALL)
        TEXT('User-defined forms control table')
```

The following is an example of adding an entry to the FCT:

```
ADDFCTE  FCT(MYRJELIB/MYRJEFCT) +
         FORMTYPE(STD) +
         DEVTYPE(*PRT) +
         LCLFORM(*FORMTYPE) +
         FILE(*WTRE) +
         MBR(*WTRE) +
         FSN*WTRE) +
         DTAFMT(*WTRE) +
         CHLVAL(*FILE) +
         FORMSIZE(*FILE) +
         LPI(*FILE) +
         CPI(*FILE) +
         COPIES(*FILE) +
         PGM(*WTRE) +
         MSGQ(*WTRE)
```

Printing Using RJE

Starting Writers

The following commands can be used to start an RJE writer:

STRRJESSN
>   Start RJE Session: Starts the writers defined in the session description (when WTRS(*YES) is specified). If this command is used, the writer options cannot be selected: each writer starts with the defaults of the Start RJE Writer (STRRJEWTR) command.

STRRJEWTR

Start RJE Writer: Starts either individual writers or all writers defined in the session description. If this command is used, individual options can be selected for each writer.

WRKRJESSN

Work with RJE Session: Allows one or more writers defined in the session description to be started. If this command is used, the RJE session status of all writers and readers defined in the session is displayed on the screen.

Cancelling Writers

If writers are cancelled with OPTION(*IMMED), or if a session ends abnormally during processing, blocks of data may be lost. Each host system has the capability to save a backup of the data. Refer to the appropriate host system manual for details.

The following commands and control statements can be used to cancel an RJE writer:

- RJE control statements

    ..CANCEL

    Cancel: Control statement that cancels a printer or punch. This control statement can be issued only from the RJE console. To issue it from the console, type it on the command line and press F10.

    ..END End: Control statement that ends the session in a controlled manner. This control statement can be issued only from the RJE console. To issue it from the console, type it on the command line and press F10.

- OS/400 commands

    ENDRJESSN

    End RJE Session: Command that cancels all writers that are active and defined in the session description.

    CNLRJEWTR

    Cancel RJE Writer: Command that cancels either individual writers or all writers that are active and defined in the session description. If this command is issued, the session does not end and the system continues to communicate with the host system.

    WRKRJESSN

    Work with RJE Session: Command that permits one or more writers defined in the session description to be cancelled.

Receiving Printer Output from the Host System

When RJE receives an output stream from the host, it uses the host-system form type value to determine how that output stream should be processed. For SNA, the form type is in the peripheral data set information record (PDIR); for BSC, it is in the forms-mount message.

The host-system form type is compared with certain values at the iSeries server, and processing continues as shown in the following figure.

Is there an FCT associated with this session?

> YES: Does the host-system form type match the FORMTYPE value in any of the FCT entries?

> > YES: Is a user program specified in the PGM parmeter of the FCT entry?

> > > YES: Call the user program.

> > > NO: What is specified in the FILE parameter of the FCT entries?

> > > > FILENAME: Direct the output to that file.

> > > > *WTRE: Get the file name from the writer entry in the session description; continue processing based on the FCT entry values.

> > > > *NONE: Use the writer entry values in the session description (not the FCT entry) to continue processing.

> NO: Is a user program specified in the PGM parameter of the writer entry in the session description?

> > YES: Call the user program.

> > NO: Direct the output to the file name specified in the writer entry of the session description; continue processing using the writer entry.

RBAFT528-0

Record Length of Output Data

In RJE, to determine the record length of output data, use the OUTRCDLEN parameter of the CRTRJECFG, CRTSSND, CHGSSND, or CVTRJEDTA command. This parameter controls how RJE reblocks the output stream received from the host. It has two values: *FIXED and *FILE.

If you specify *FIXED, RJE reblocks the output stream according to its own defaults, which are as follows:

- When an RJE printer is processing the output stream:
  - If data is being written to a printer file, the record length is 132 bytes for 10 or fewer characters per inch (CPI), and 198 bytes when the CPI is greater than 10. You must specify *FCFC for the CTLCHAR parameter of either the Create Printer File (CRTPRTF) or Change Printer File (CHGPRTF) command.
  - If data is being written to a physical file, the record length is 132 bytes for data format, and 133 bytes for FCFC format. Physical files do not have any CPI value associated with them. The record length of the physical file must equal the record length of the output data (132 or 133 bytes) to prevent record misalignment.
- RJE punches reblock data as 80-byte records.

- For diskette files, the output stream is reblocked as 128-byte records. If the record length is greater than 128, data is truncated.

If you specify *FILE, RJE reblocks the output stream as follows:
- When an RJE printer is processing the output stream:
  - If data is being written to a printer file, the record length is determined by the page width of the printer file. You must specify *FCFC for the CTLCHAR parameter of either the Create Printer File (CRTPRTF) or Change Printer File (CHGPRTF) command.
  - If data is being written to a physical file, the record length of the output data is determined by the record length of that file. If the data is FCFC format, the record length of the file must include one extra byte for the FCFC code.
- When an RJE punch is processing the output stream:
  - If data is being written to a printer file, the record length is determined by the page width of the printer file.
  - If data is being written to a physical file, the record length of the output data is determined by the record length of that file.
- For diskette files, the output stream is reblocked as 128-byte records. If the record length is greater than 128, data is truncated.

The maximum record length that RJE supports, including carriage-control characters, is 255 bytes. To ensure record alignment, the record length of the output file must be greater than or equal to the logical record length of the data that is received from the host subsystem. If the record length of the output file is greater than the logical record length of the data received, the records are right-padded with blanks.

If you need to use both OUTRCDLEN(*FILE) and OUTRCDLEN(*FIXED) in one session, specify the former and then have it function as OUTRCDLEN(*FIXED) by setting the record length of the output files to 80, 132, 133, or 198.

Example of Using OUTRCDLEN(*FILE)
Assume that you want to send a report that has a record length of 144 bytes from a host to an iSeries server. To do this using OUTRCDLEN(*FILE):

1. Create a physical file with a record length of 144 bytes using either the Create Physical File (CRTPF) or the Create Source Physical File (CRTSRCPF) command. Specify *NOMAX for the MAXMBRS parameter to allow RJE to create as many members as needed in the file.
2. Change the writer or FCT entry FILE parameter to specify the new physical file. When the data is written to the file, it is reblocked to 144-byte records.

Printing Using FCFC

The host system sends output streams as 132- or 198-byte records with carriage-control characters that go before each record. RJE converts records containing host system carriage-control characters to records containing ANS FCFC codes (before printing) that are supported by the OS/400 program.

For SNA, carriage control is provided by the extended SNA character string (SCS) controls. These are shown in Table 32 on page 335.

Table 32. SNA Character String (SCS) Controls

SCS	Meaning
CR	Carriage return
FF	Forms feed (page end)
IRS	Inter-record separator
NL	New line
SVF	Set vertical format
VCS	Vertical channel set

For BSC, carriage control is provided by carriage-control bytes (CCBs). These are shown in Table 33.

Table 33. BSC Carriage-Control Bytes

Code	Meaning
1010 00nn	Space immediately nn spaces (no printing)
1011 nnnn	Skip immediately to channel nnnn (no printing)
1000 00nn	Space nn lines after printing
1001 nnnn	Skip to channel nnnn after printing
1000 0000	Suppress space after printing

The ANS FCFC codes are shown in Table 34.

Table 34. ANS First-Character Forms Control (FCFC) Codes

Code	Meaning
	Space 1 line before printing (blank code)
0	Space 2 lines before printing
-	Space 3 lines before printing
+	Suppress space before printing
1	Skip to channel 1
2	Skip to channel 2
3	Skip to channel 3
4	Skip to channel 4
5	Skip to channel 5
6	Skip to channel 6
7	Skip to channel 7
8	Skip to channel 8
9	Skip to channel 9
A	Skip to channel 10
B	Skip to channel 11
C	Skip to channel 12

When you specify *FIXED for the OUTRCDLEN parameter of the CRTRJECFG, CRTSSND, CHGSSND, or CVTRJEDTA command, the output stream is reblocked as follows:

- When an RJE printer is writing the output data to a printer file, the record length is 132 bytes for 10 or fewer cpi, and 198 bytes when the cpi is greater than 10. You must specify *FCFC for the CTLCHAR parameter of either the Create Printer File (CRTPRTF) or Change Printer File (CHGPRTF) command.
- When an RJE printer is writing the output data to a physical file, the record length is 132 bytes for data format, and 133 bytes for FCFC format. Physical files do not have any cpi value associated with them. The record length of the physical file must equal the record length of the output data (132 or 133 bytes) to prevent record misalignment.

When you specify *FILE for the OUTRCDLEN parameter, the record length (page width) of the file that receives the data determines the record length of the output data.

Using a User Program to Receive Host-System Output

Three programs supplied with RJE can be called by user-written programs to receive data from the host system. See Chapter 8 of the Remote Job Entry (RJE) Guide.

3x74 Attached Printers

There is no 3270 emulation pass-through (*pipeline*) support for 3x74 remote-attached printers. Printing capability is limited to the LU 1 support. Printer data management and spool support is not provided for printers that use the 3270 Information Display System data-stream capability (DSC).

For OS/400 printing applications, the extent of the function is limited to that provided by a 3287 printer, regardless of the type and model of the printer actually attached. This is functionally equivalent to a 5256-003 printer. For example, a 4224 printer, which has all-points-addressable capability, only provides 3287 printer capability when attached to a 3x74 controller. However, if a 3284 printer is attached to the 3x74 controller, it does not provide the 6 or 8 LPI support of the 3287 printer because it is not equipped with the necessary hardware features.

When using a 3x74 remote-attached printer (for example, a 4224) configured as a 3287 printer, make sure that the page length set on the printer hardware switch matches the length of the forms that are in the printer. If the length of the forms loaded does not match the printer hardware switch setting, the application will still print correctly on the forms. However, when the spooled file completes printing, the 4224 uses the default page size set on the printer hardware switches.

For example, assume your page length set on the printer hardware switches is 11 inches, your actual forms are 3 inches long, the spooled file just completed printing, and you press the Page Eject key to get the last printed form. By pressing the Page Eject key, you may advance the forms 11 inches instead of the 3 inches your form takes up.

DBCS Printer Considerations

If the spooled file contains double-byte data (IGCDTA(*YES) specified on the printer file), it can be printed on printers attached to a 3x74 controller.

However, the following attributes or DDS keywords are ignored if the target printer is attached to a 3x74 controller.

- IGCCPI (DBCS attribute)
- IGCCHRRTT (DBCS attribute and DDS keyword)
- IGCEXNCHR (DBCS attribute)
- IGCCHRSIZ (DDS keyword)
- DFNLIN (DDS keyword)

The IGCSOSI printer file parameter is processed in the OS/400 program and not in the printer being used to print the spooled file.

Distributed Data Management (DDM) Printing

DDM enables users or application programs on the iSeries server to access data files that reside on remote systems and allows these remote systems to access data on the local system. This remote file access is transparent to the application program command or utility. Remote file requests are routed through a DDM file, which contains the name of the file on the remote system as well as the name of this system as it is known on the network.

There are, however, certain limitations to the DDM support, particularly where printing applications are concerned. For example:

- The iSeries Query licensed program does not support DDM files.
- It is not possible to print on a remote iSeries server using, for example, an OVRPRTF of QSYSPRT to the source DDM file because the print request is rejected by DDM, as shown by the following job log extract:

```
3 > OVRPRTF FILE(QSYSPRT) TOFILE(ITSCID03/DDMQSYSPRT)
DDM file DDMSRC in MYLIB uses remote file QSYS/QSYSPRT.
File DDMSRC in MYLIB not a data base file.
Cannot open DDM file DDMSRC in MYLIB.
Function check. CPF4207 unmonitored by QWSGET at statement *N,
instruction X'0F9D'.
A function check was received while opening the print file.
The print operation used the default printer device file.
```

One way in which data could be printed on a remote iSeries server using DDM is as follows:

1. Create a DDM file on the source system using the Create DDM File (CRTDDMF) command.

   ```
   CRTDDMF FILE(MYLIB/DDMSRC) RMTFILE(QSYS/QSYSPRT)
   RMTLOCNAME(B20) TEXT('DDM file for remote printing on B20').
   ```

2. Create a physical file of the appropriate record length to receive the spooled data (80 bytes for print screen output, 132 bytes for normal spooled output), using the Create Physical File (CRTPF) command.

   ```
   CRTPF FILE(MYLIB/PFILE) RCDLEN(80) TEXT('CPSPLF data file').
   ```

3. Create a similar file on the remote iSeries server using the Submit Remote Command (SBMRMTCMD) command.

   ```
   SBMRMTCMD CMD('crtpf file (rlib/rfile) rcdlen(80)') DDMFILE(DDMSRC).
   ```

4. Copy the spool entry to the physical file using the Copy Spooled File (CPYSPLF) command, taking the defaults (no control character).

   ```
   CPYSPLF FILE(QSYSPRT) TOFILE(MYLIB/PFILE)
   JOB(003049/USER/DSP06).
   ```

5. Use the Copy File (CPYF) command to copy the data from this physical file to the remote physical file created in Step 3.

```
CPYF FROMFILE(MYLIB/PFILE) TOFILE(MYLIB/DDMSRC)
MBROPT(*ADD).
```

6. Use the SBMRMTCMD command once again to copy the data from the remote physical file to QSYS/QSYSPRT in order to create a spooled file on the remote iSeries server's default output queue.

```
SBMRMTCMD CMD('cpyf fromfile(rlib/rfile) tofile(qsysprt) mbropt(*add)')
DDMFILE(DDMSRC).
```

This procedure can be adapted to direct OS/400 printing to remote System/36 systems and System/38 systems with DDM installed. See the Distributed Data Management book for details of the differences in DDM implementation on these other systems.

Object Distribution Printing

Object distribution provides the facility to send objects, messages, job streams, and spooled files across a suitably configured SNA link to another iSeries server or group of servers. File distribution (data, source, and spooled files) to System/36, System/38, and System/390 CICS* systems is also possible using OS/400 object distribution support. See the SNA Distribution Services book for details.

Chapter 18. The IBM Internet Printing Protocol (IPP) server for iSeries

This chapter provides information about the IBM Internet Printing Protocol (IPP) Server for iSeries and how to enable it to provide additional support for printers that are attached to an iSeries server.

What is the Internet Printing Protocol?

The Internet Printing Protocol (IPP) defines an industry standard method of delivering print jobs using Internet technologies. The IPP protocol was developed by the Printer Working Group, a consortium of the major companies involved in network printing.

IPP is transported over HTTP/1.1 using a message body whose content-type is "application/ipp". The well-known port 631 has been assigned to IPP.

Why use the IPP server?

IPP provides a single standard interface for submitting print jobs to iSeries servers. IPP works on any TCP/IP network, so you can use the same process to submit print jobs on your local area network, your intranet, or the Internet.

IPP allows business travelers, telecommuters, or anyone working remotely to submit and manage print jobs on a remote iSeries server.

What is supported by the IPP server?

The IPP server supports IPP version 1.1. The following required operations are supported:
- Print-job
- Validate-job
- Get-printer-attributes
- Get-jobs
- Cancel-job
- Get-job-attributes

Additionally, the following optional IPP operations are supported:
- Pause-printer
- Resume-printer
- Purge-jobs
- Hold-job
- Release-job
- Restart-job

The IPP server can be configured to provide security features for user authentication and encryption of print data.

Setting up the IPP server

You can administer and configure the IPP server using the IBM IPP Server Administrator for iSeries, a browser-driven graphical user interface. Before using the interface, verify that the IBM HTTP Server for iSeries is installed on your system. For upgrade to TLS, or SSL support, you must also have installed the Digital Certificate Manager and a Cryptographic Access Provider licensed program.

Setting up your Internet browser

To allow the administrator interface to function correctly, JavaScript™ must be enabled for your Internet browser. This can usually be enabled by accessing the preferences or options feature of your browser.

To select the display language for the administrator, access the language selection for your Internet browser, and select or enter a supported locale that is installed on your system. Normally, there is a list to choose from. If a list is not displayed, you can manually enter a locale. The locale is usually a two letter abbreviation that represents a language. For example, English is represented by the letters "en". Sometimes the locale may further be defined by a country or region. For example, French spoken in Switzerland is represented by "fr_CH". If the locale defined for the browser is not installed or supported by the administrator interface, the language associated with the user profile is used. If this is language is not supported, English is used by default.

Using the Administrator Interface

To access the administrator interface, verify that the IBM HTTP Server for iSeries is installed and the *ADMIN server instance has been started. This is done so that your Internet browser can access the iSeries Tasks page.

To start the *ADMIN server instance, use the following command:

```
STRTCPSVR SERVER(*HTTP) HTTPSVR(*ADMIN)
```

Use the following URL to access the iSeries Tasks page:

```
http://system:2001
```

where *system* is the name of the iSeries server. From the iSeries Tasks page, click on the icon for the IBM IPP Server.

You should now be presented with the IBM IPP Server Administrator interface. To use the administrator interface, you must have *IOSYSCFG authority, and you must have read and write access to the following property files:

- QIBM/UserData/OS400/Ipp/conf/qippsvr-cust.conf
- QIBM/UserData/OS400/Ipp/conf/printer.properties

In order to change preferences, you must have read and write access to the preference property file, QIBM/UserData/OS400/Ipp/conf/preferences.properties, if it exists.

From the IBM IPP Server Administrator interface, you can perform the following tasks:

- Configure the IPP server
- Display, create, change, and delete IPP printer configurations
- Set up preferences for using the browser interface
- Manage the IPP server

The administrator interface is divided into two main sections, a navigational frame on the left, and a form frame on the right. The form frame is further broken down into a form content frame, a button frame, and a message frame. The message frame, located at the bottom of the form frame, is the location where messages appear during normal operation of the browser interface.

To change navigational menus, use the navigation tabs located above the frames near the top of the browser window. The choices available are:

Administration
> Provides a form for managing the IPP server.

Configuration
> Provides forms for configuring the IPP server, including SSL, defining access log and error log protocol, and working with IPP printer configurations.

Other Links
> Provides links to related Web sites.

There are two document links at the top of the browser interface as well. Each of the document links will display the selected document in a separate browser window. The document links are:

Getting Started
> Provides information on using the interface.

Printer Device Programming Guide
> Links directly to the online version of the Printer Device Programming book.

Configuring the IPP server

If SSL connections are desired for any IPP printer, you must first enable the IPP server to use SSL by specifying an SSL port. To enable SSL connections:

1. Click on the **Configuration** tab to load the configuration navigational menu.
2. Click on **Basic Configuration** to load the basic configuration form.
3. Select the **Enabled** radio button.
4. Fill in a valid port number in the **SSL Port** text field.
5. Click the **Apply** button.

The IPP server is automatically configured to port 631 for non-SSL data. To support upgrade to TLS on port 631, or to support SSL on a specified port, a digital certificate must be associated with the IPP server. To do this, use Digital Certificate Manager for iSeries (DCM) and associate a certificate with the IPP server, listed as server application QIBM_IPP_QIPPSVR.

To disable SSL connections, select the **Disabled** radio button on the form and click the **Apply** button.

Note: Whenever a change is made to the configuration of the IPP server, whether it is for SSL connections or logging information, the IPP server must be stopped and then started again for the change to become effective.

To configure how you want access logs created, written, and archived, follow these steps:

1. Click on the **Configuration** tab to load the configuration navigational menu.

2. Expand the **Logs and Errors** menu item.
3. Click on **Access Logs** to load the access logs form.
4. Fill in the form with the appropriate information.
5. Click the **Apply** button.

To configure how you want error logs created, written, and archived, perform the following steps:

1. Click on the **Configuration** tab to load the configuration navigational menu.
2. Expand the **Logs and Errors** menu item.
3. Click on **Error Logs** to load the error logs form.
4. Fill in the form with the appropriate information.
5. Click the **Apply** button.

Note: The IBM IPP Server Administrator and the IBM IPP Server are two different applications. However, error logging must be enabled for the IPP server to allow administrator interface error logging to be enabled. If an error occurs using the IPP server, check the IPP server error log, specified using the error log form of the administrator interface, to help locate the problem.

Creating an IPP Printer Configuration

To create an IPP printer configuration do the following:

1. Click on the **Configuration** tab to load the configuration navigational menu.
2. Expand the **Internet Printers** menu item.
3. Click on **Create configuration** to start the create task wizard.
4. Follow the task steps to create an IPP printer configuration.
5. Click the **Finished** button on the confirmation panel to create the configuration.

Changing an IPP Printer Configuration

To change an IPP printer configuration do the following:

1. Click on the **Configuration** tab to load the configuration navigational menu.
2. Expand the **Internet Printers** menu item.
3. Click on **Change configuration** to start the change task wizard.
4. Follow the task steps to change an IPP printer configuration.
5. Click the **Finished** button on the confirmation panel to change the configuration.

Viewing an IPP printer configuration

To view all of the IPP printers configured on the system:

1. Click on the **Configuration** tab to load the configuration navigational menu.
2. Expand the **Internet Printers** menu item.
3. Click on **Display Configurations**.

Deleting an IPP printer configuration

To delete an IPP printer configuration do the following:

1. Click on the **Configuration** tab to load the configuration navigational menu
2. Expand the **Internet Printers** menu item.
3. Click on **Delete configuration** to display a list of configured IPP printers.
4. Select the IPP printer to delete.

5. Click the **Delete** button.

Managing the IBM IPP server

Using the Manage IBM IPP Server form, you can use the start, stop, and restart buttons to perform the given action for the IPP server. The current server status, along with the ports the server is listening on, is displayed in the table. At most, there will be two ports the IPP server will be listening on at any one time. Port 631 is for non-secure data and secure data provided by a connection upgrade to TLS. The other port is for secure data provided by an SSL connection. The last time the table information was updated is also displayed. The table can be refreshed by clicking the refresh button.

To manage the IPP server, do the following:

1. Click on the **Administration** tab to display the administration navigational menu.

2. Click on the **Manage IBM IPP Server** menu item.

The IPP server is initialized using a special configuration file. The file is named `QIBM/UserData/OS400/Ipp/conf/qippsvr-cust.conf`. Do not manually edit this file. The configuration file is changed via the IBM IPP Server Administrator browser interface.

Troubleshooting

If an error occurs using the IBM IPP Server Administrator, the message frame box will usually contain the error data and provide information on how to correct the problem. For additional information, internal tracing can be used. Tracing provides a mechanism for retrieving state information during execution of the administrator and the IPP server. To activate tracing, specify a valid error log file name using the **Error Logs** form. Then, select a logging level below **Critical**. Selecting a logging level of **Debug** will provide the maximum amount of tracing information. The trace log containing detailed information on the operation of the administrator is located in file /QIBM/UserData/OS400/Ipp/Logs/qippcfg.log. The trace log containing detailed information on the operation of the IPP server is located in file /QIBM/UserData/OS400/Ipp/Logs/qippsvr.log.

To stop internal tracing for the administrator and the IPP server, select a logging level above "Error" (ie, Critical, Alert, or Emergency) on the **Error Logs** form and click the **Apply** button.

Note: The trace files /QIBM/UserData/OS400/Ipp/Logs/qippcfg.log and qippsvr.log are deleted every time tracing/error logging is started. If you wish to trace, it is recommended that at regular intervals you stop tracing, archive the trace file, and start the tracing again.

Use the following table to help troubleshoot other problems you may encounter working with the IBM IPP Server Administrator or the IBM IPP Server.

Table 35. Troubleshooting

You do not have the correct authority to a file.	To modify any configuration files, IBM IPP Server Administrator browser interface users require at least *IOSYSCFG authority. Verify that you have the correct authority.

Table 35. Troubleshooting (continued)

File does not exist or is corrupt.	Always make sure the following files exist on your system: /QIBM/UserData/OS400/Ipp/conf/ qippsvr.conf /QIBM/UserData/OS400/Ipp/conf/ preferences.properties /QIBM/UserData/OS400/Ipp/conf/ printer.properties You should not manually edit these files unless you know exactly what you are doing. Modifying one file may require you to modify additional files. Manually changing these files could result in failure of server initialization.
The log file name is not valid.	Click the "?" help icon for the log file name field for more information on valid log file names.
You made changes to the IPP server configuration, but they do not appear to be recognized.	Stop the IPP server and start it again. Whenever a change is made to the server's configuration, the IPP server needs to be restarted in order for the change to become effective.
The browser interface is not being displayed in the language specified.	The language specified in the browser is not supported by the IBM IPP Server Administrator browser interface application.
You cannot find help for a field.	Click the "?" help icon for information concerning a particular field. You may also activate "verbose" for the form, which provides a more detailed overview of the entire form.
An internal error occurred using the browser interface.	Turn on tracing and examine the trace file to help locate the problem.
The IPP server does not start or stay running after you attempt to start it.	An error has probably occurred during initialization. Whenever the IPP server is started, an OS/400 fully-qualified job name is displayed in the message frame, in the form: JobNumber/JobUser/JobName If the IPP server fails to start do the following: 1. Log on to the iSeries server. 2. WRKJOB JOB(JobNumber/JobUser/JobName) . 3. Select option 4, "Work with Spooled Files". 4. Using display option 5, view any spooled files to help locate the cause of the problem. If you have manually edited the IPP server configuration file, a newly specified directive may be causing the IPP server to fail initialization. If this is the case, edit the configuration file and remove the directive in error.

Table 35. Troubleshooting (continued)

The IPP server does not stop after you attempt to stop it.	An error has occurred on the iSeries server. To stop the IPP server manually: 1. Log on to the iSeries server. 2. WRKACTJOB SBS(QHTTPSVR). 3. Find the QIPPSVR jobs. 4. Use option 4 and end the jobs immediately.
When SSL is enabled, the IPP server does not stay running.	The IPP server may not have a valid certificate associated with it for SSL connections. Use Digital Certificate Manager for iSeries to create a certificate and associate it with the IPP server. (The IPP server itself should be registered with DCM automatically.) Associate the certificate with the application `QIBM_IPP_QIPPSVR`.

Part 6. Appendixes

Appendix A. Examples of Working with Printing Elements

The examples in this appendix are intended to help you become familiar with the printing elements used to perform printing tasks.

Any one element by itself will probably not allow you to control your printing environment. However, by becoming familiar with all the elements and how they interact, you will be able to create and control your printing environment.

Structure of Examples in This Appendix

Most of the examples assume that spooled files have been created and are in an output queue. If you want to quickly create a spooled file to use with these examples, you can press the Print key. Pressing the Print key creates a spooled file containing the data shown on the display at the time the Print key was pressed.

Throughout the examples, the use of CL commands and F4 (Prompt) are emphasized as a way to work with all the elements that make up printing.

Type of Example	Where to Find It
Working with your user profile	To see how to display and change your user profile, go to "Working with Your User Profile".
Working with QPRTDEV system value	To see how to use the system value QPRTDEV to find out what the default system printer (printer device) is, go to "Working with System Values" on page 350.
Creating output queues	To see how to create output queues, go to "Working with Output Queues" on page 351.
Working with printer files	To see how to specify different output queues and printer devices by changing printer file attributes, go to "Using the Printer File to Select a Different Output Queue" on page 352
Moving spooled files	To see how to move spooled files from one output queue to a different output queue, go to "Moving a Spooled File to a Different Output Queue" on page 353.
Assigning printers to output queues	To see how to assign a printer from one output queue to a different output queue, go to "Working with Printer Writers (WRKWTR)" on page 355.
Locating spooled files	To see how to locate spooled files created by you, go to "Locating Spooled Files" on page 357.

Working with Your User Profile

Everyone who uses the system has a user profile; information stored in the user profile will help you with your printing needs.

These commands allow you to display and change your user profile:
- DSPUSRPRF (your user ID) allows you to display your user profile.
- CHGUSRPRF (your user ID) allows you to change your user profile.

Displaying Your User Profile

From a command line, type DSPUSRPRF (your user ID) and press the Enter key.

Your user profile appears. From a printing perspective, the important parameters are: *Output queue* and *Printer device*. Page through your displayed user profile and locate the output queue and printer device parameter values. These values indicate where your spooled files go (the output queue) and which printer (the printer device) they are printed on.

When you display your user profile, you cannot change any of the attributes you are viewing.

Changing Your User Profile

Let's say that your organization purchased a printer that supports letter-quality printing as compared to the dot-matrix printer you currently use. You want your output to be printed on the new, letter-quality printer.

Your system administrator has configured the new printer and created an output queue for it.

The printer name is PRT99 and its output queue name is also PRT99.

Different Methods to Change Your User Profile
There are two ways to change your user profile to reflect the new printer's name and output queue.

Method one: From a command line, type CHGUSRPRF (your user ID), and press F4 (Prompt).

Your user profile with all its assigned values appear. Page through the user profile until you find the output queue and printer device parameter values. Type PRT99 over the current values, and press the Enter key.

Method two: From a command line, type

```
CHGUSRPRF (your User ID) OUTQ(PRT99) DEV(PRT99)
```

Press the Enter key.

Verification of Change
To verify that the requested change took place (regardless of the method of change) use the Display User Profile command DSPUSRPRF (your user ID). Page forward to view the output queue and printer device parameter values. They will be PRT99.

Note: When you change your user profile, the change does not become effective until the next time you sign on.

Working with System Values

The QPRTDEV system value determines which printer is defined as the system printer.

Displaying the System Value for the System Printer

To find the name of the system printer, display the system value called QPRTDEV. Type DSPSYSVAL QPRTDEV and press the Enter key. The following is displayed:

```
                              Display System Value

System value . . . . . :    QPRTDEV
Description  . . . . . :     Printer device description



Printer device . . . . :    PRT01          Name
```

You can see, from the displayed system value QPRTDEV, that the name of the
printer your output goes to is PRT01 (if the SPOOL parameter of the printer file is
*NO).

If the SPOOL parameter of the printer file is *YES, your spooled files go to output
queue PRT01.

Note: This is true only if the default values for the printer file, job description, and
user profile have not been changed.

Changing System Values

Very few users have the authority to change system values.

If system values are changed, everyone on the system could be affected.

In this case, if you are using *SYSVAL for the output queue and printer device
values in your user profile and someone changes the QPRTDEV value, your
printed output could go to a different output queue or printer.

Working with Output Queues

Output queues can be created by any user of the system.

You might choose to create additional output queues if you have application
programs that have special forms requirements, limited number of printers
available on the system, low speed printers, or spooled files that you do not want
printed. Creating additional output queues to hold spooled files that require
special handling allows you to determine what gets printed, on what printer, and
when it gets printed.

Until you become very familiar with using CL commands to create output queues,
the best way to start is to type CRTOUTQ and press F4 (Prompt). The following
display appears:

```
                        Create Output Queue (CRTOUTQ)
Type choices, press Enter.
Output queue . . . . . . . . . .    XXXXXXXXX     Name
  Library  . . . . . . . . . . .      *CURLIB     Name, *CURLIB
Order of files on queue  . . . .    *FIFO         *FIFO, *JOBNBR
Text 'description' . . . . . . .    *BLANK
                          Additional Parameters
Display any file . . . . . . . .    *NO           *NO, *YES
Job separators . . . . . . . . .    0             0-9, *MSG
Operator controlled  . . . . . .    *YES          *YES, *NO
Authority to check . . . . . . .    *OWNER        *OWNER, *DTAAUT
Authority  . . . . . . . . . . .    *USE          *USE, *ALL, *CHANGE, *EXCLUDE
```

The only required value is for the *Output queue* parameter. The name you assign here is the name you will use when specifying this output queue in a printer file, job description, or user profile.

The remainder of the output queue parameters have a default value assigned.

For detailed information on the CRTOUTQ command and its parameters, press the Help key or see the CL Reference information in the iSeries Information Center.

Using the Printer File to Select a Different Output Queue

An output queue is an object that holds your spooled files until they are printed. Printer files, job descriptions, and user profiles all have parameters that allow you to specify a particular output queue.

In this example, let us assume that an application program that is run every day uses a printer file, called PRTF1, to control which output queue your spooled files are sent to.

In the printer file named PRTF1, the output queue value is OUTQ1 and the printer device value is PRT3812.

Let us also assume that PRT3812 is out of operation for an unknown amount of time. You need to print the spooled files from your application program but will have to use a printer named PRT3816 and an output queue named OUTQ3816. How can you change printer file PRTF1 to have your spooled files printed on PRT3816?

There are two ways, one temporary and one permanent, that you can send your spooled files to PRT3816.
- Temporarily, using the Override Printer File (OVRPRTF) command.
- Permanently, using the Change Printer File (CHGPRTF) command.

Using the OVRPRTF Command to Change Printer File PRTF1
The Override Printer File (OVRPRTF) command changes PRTF1 only for the job, program, or display session in which it is issued.

In our example,
1. Type OVRPRTF (PRTF1) and press F4 (Prompt).
2. Page forward until you locate the output queue and printer device parameters.
3. Type over the existing values for output queue and printer device with OUTQ3816 and PRT3816 respectively.

   Note: At this point the spooled files will go to OUTQ1 if the SPOOL parameter in the printer file is set to *YES. The output will go directly to PRT3812 if the SPOOL parameter in the print file is set to *NO.
4. Press the Enter key.

The application program that uses printer file PRTF1 will now send the spooled files to output queue OUTQ3816.

Using the CHGPRTF Command to Change Printer File PRTF1
The Change Printer File (CHGPRTF) command changes PRTF1 permanently and will be in effect as soon as the command is run by the system.

When a change is made to the printer file, it affects all users who use that particular printer file.

In our example,

1. Type CHGPRTF (PRTF1) and press F4 (Prompt).
2. Page forward until you locate the output queue and printer device parameters.
3. Type over the existing values for output queue and printer device with OUTQ3816 and PRT3816 respectively.
4. Press the Enter key.

The application program that uses printer file PRTF1 will now send the spooled files to output queue OUTQ3816.

Moving a Spooled File to a Different Output Queue

Notes:

1. Security measures enforced by the system protect the spooled files on the iSeries server. You may discover that you cannot move some spooled files to some output queues. Contact your security officer if this happens. You can move *your* spooled files to other output queues. You cannot move *someone else's* spooled files to other output queues unless you have certain authorities. Contact your security officer for specific details, or see Basic system security and planning under the Security topic in the iSeries Information Center.
2. When moving spooled files to other output queues, you should keep in mind that, if a printer is not assigned to the target output queue, you may have to assign a printer to the output queue.

Different Methods to Move Spooled Files

Here are three methods to move spooled files from one output queue to another output queue.

1. Using the Work with Spooled Files (WRKSPLF) command
2. Using the Work with Output Queue (WRKOUTQ) command
3. Using the Work with Job (WRKJOB) command

Note: If you choose to move spooled files from one output queue to another using CL commands from a command line, you must know the library the target output queue is in.

Method one: Moving spooled files using the WRKSPLF command.

The Work with Spooled Files (WRKSPLF) command can be used to move a spooled file from one output queue to another:

1. Type WRKSPLF and press the Enter key.

   The Work with All Spooled Files display appears. This display lists all of your spooled files and shows you what output queue they reside in.
2. Move the cursor to the spooled file you want to move and type a 2 (Change) in the option field next to the file name.

   This allows you to change the attributes of this spooled file. In this example, the attribute you will change is the output queue name (and library if the output queue is in a different library).
3. Press the Enter key and the Change Spooled File Attributes (CHGSPLFA) display appears.

4. Move the cursor to the *Output queue* parameter, and type over the current output queue name with the name of the output queue you want the spooled file moved to.

   If the library of the target output queue is different from the library that your current output queue is in, type the name of the library that your target output queue is in.

5. Press the Enter key, and you return to the Work with All Spooled Files display.

   The name of the target output queue is now shown in the *Queue* column next to the spooled file that you moved.

Method two: Moving spooled files using the WRKOUTQ command.

The Work with Output Queue (WRKOUTQ) command can be used to move a spooled file from one output queue to another:

1. Type WRKOUTQ and press the Enter key

   The Work with all output queues display appears.

2. Page forward until you find the output queue that your spooled files go to.

3. Type a 5 (Work with) next to the output queue you want to work with.

   The Work with output queue display appears.

4. Page forward until you locate the spooled file you want to move to a different output queue.

5. Move the cursor to the spooled file and type a 2 (Change) in the option field next to the file name.

   This allows you to change the attributes of this spooled file. In this example, the attribute you will change is the output queue name (and library if the output queue is in a different library).

6. Press the Enter key and the Change Spooled File Attributes (CHGSPLFA) display appears.

7. Move the cursor to the *Output queue* parameter and type over the current output queue name with the name of the output queue you want the spooled file moved to.

   If the library of the target output queue is different from the library that your current output queue is in, type the name of the library that your target output queue is in.

8. Press the Enter key, and you return to the Work with Output Queue display.

   The value in the status (Sts) column is now *CHG. The spooled file is now in the target output queue.

Method three: Moving spooled files using the WRKJOB command.

The Work with Job (WRKJOB) command can be used to move any spooled file that was created by the present job from one output queue to another:

1. Type WRKJOB and press the Enter key.

   The Work with Job display appears.

2. Select option 4 (Work with spooled files).

   The Work with Job Spooled Files display appears.

3. Page forward until you locate the spooled file you want to move to a different output queue.

4. Move the cursor to the spooled file and type a 2 (Change) in the option field next to the file name.

This allows you to change the attributes of this spooled file. In this example, the attribute you will change is the output queue name (and library if the output queue is in a different library).

5. Press the Enter key and the Change Spooled File Attributes (CHGSPLFA) display appears.

6. Move the cursor to the *Output queue* parameter and type over the current output queue name with the name of the output queue you want the spooled file moved to.

   If the library of the target output queue is different from the library that your current output queue is in, type the name of the library that your target output queue is in.

7. Press the Enter key, and you return to the Work with Job Spooled Files display.

   The name of the target output queue is now shown in the *Queue* column next to the spooled file that you moved.

Working with Printer Writers (WRKWTR)

The WRKWTR command allows you to control the printers that are configured to your system. When you type WRKWTR and press the Enter key, the Work with All Printers display appears. From this display you can:

- Start a writer.

  When this option is selected, the printer writer program tells the printer to start printing spooled files from the output queue it is assigned to.

- Change a writer.

  When this option is selected, you can change some of the attributes of the printer. For example: form type, number of file separator pages, or output queue the printer is assigned to.

- Display the messages for the writer.

  When this option is selected you can view messages that are returned to the person working with a particular printer. Answering messages is important to keep the printer in an active state.

- Hold a writer.

  When this option is selected, the printer writer program tells the printer to stop printing.

- Release a writer.

  When this option is selected, the printer is again available for use.

- End a writer.

  When this option is selected, the printer stops and becomes the printer available to the system for use. For example: the printer can now be assigned to print spooled files from a different output queue.

Assigning a Printer to a Different Output Queue

This example shows how to assign a printer to print spooled files from an output queue the printer is not currently assigned to.

Assumptions for This Example

1. Assume you have an output queue named MYOUTQ.

2. Assume it is not currently assigned to any printer.

3. Assume your system has a printer named PRT3816 and it is assigned to an output queue also named PRT3816. PRT3816 is the printer you want assigned to the output queue named MYOUTQ.

Different Methods to Assign Printers to Output Queues

1. Using the displays.

   This is the method you would use if you are not familiar with CL commands.

2. Using a CL command.

   This is the method you would use if you are familiar with CL commands and are familiar with the elements that help you manage your printing work load.

Method one: Before you can assign a printer to print spooled files you have to know the status of the printer you want to use.

1. To find the status of the printer you want to use, type WRKWTR on any command line and press the Enter key. The Work with All Printers display appears. This display shows all printers configured to the system.

2. Page forward until you find PRT3816.

   Look at the Sts (status) column to determine the status of the printer. Let us assume it is STR (started).

3. To find out which output queue it is assigned to, type an 8 (Work with output queue) next to PRT3816 and press the Enter key.

   The Work with Output Queue display appears. You can now see the name of the output queue PRT3816 is assigned to and if there are any spooled files in that output queue.

   Note: If there are spooled files in the output queue and you assign the printer to a different output queue, the spooled files in PRT3816 will not print. You should notify the owner of those spooled files if you assign the printer to a different output queue.

   Let us assume there are no spooled files in the output queue assigned to PRT3816.

4. Press F12 to return to the Work with All Printers display.

5. Type a 2 (Change) next to PRT3816 and press the Enter key. The Change Writer display appears.

6. Move the cursor to the *output queue* parameters.

7. Type the name of the output queue you want to use (in this case, MYOUTQ).

   The writer will start processing spooled files from MYOUTQ.

8. Press the Enter key.

   You are returned to the Work with All Printers display.

9. Press F5 (Refresh).

10. Type an 8 (Work with output queue) next to PRT3816 and press the Enter key.

    The Work with Output Queue display appears. At the top of the display, you will see `Queue: MYOUTQ`.

    This verifies that printer PRT3816 is now assigned to the output queue named MYOUTQ.

11. Press F12 to return to the Work with All Printers display.

Method two: Using a CL command.

From a command line type:

```
CHGWTR WTR(PRT3816) OUTQ(MYOUTQ)
```

This command assigns printer PRT3816 to print spooled files from output queue MYOUTQ.

Locating Spooled Files

Most application programs create spooled files. Depending on the tasks you are performing, your spooled files could go to different output queues. Which output queues are used depends on the element used to direct the spooled files. For example: printer file, job description, or user profile can be the element that specifies which output queue is used.

Using the Work with Spooled Files (WRKSPLF) Command

To display *all* of your spooled files:

1. Type WRKSPLF and press the Enter key.

   The Work with All Spooled Files display appears.

2. Page forward to see all of <u>your</u> spooled files.

   The Device or Queue column shows you which output queue a particular spooled file is in.

Using the Work with Job (WRKJOB) Command

The difference between the WRKSPLF command and the WRKJOB command is that the WRKJOB command displays spooled files from the current sign-on session only.

To display all of your spooled files for this sign-on session:

1. Type WRKJOB and press the Enter key.

   The Work with Job display appears.

2. Select option 4, Work with spooled files.

   The Work with Job Spooled Files display appears.

   The Device or Queue column shows you which output queue a particular spooled file is in.

Options You Can Select Using WRKSPLF or WRKJOB

From the Work with All Spooled Files display or Work with Job Spooled Files display you can:

- Send a spooled file to another system through the network (option 1)
- Change spooled file attributes (option 2)
- Hold spooled files (option 3)
- Delete spooled files (option 4)
- Display the contents of a spooled file (option 5)
- Release the spooled file if it was previously held (option 6)
- Display the messages that prevent the spooled file from printing (option 7)
- View the attributes of a spooled file (option 8)
- Work with the printing status of a spooled file (option 9)

Appendix B. CL Commands Frequently Used While Working with Printing Tasks

Below are lists of CL commands with a short description of each command. You can copy them for your use.

These commands can be entered with the required parameters from any command line, you can enter the command and press F4 to use the command prompt display.

Commands Used with a User Profile

WRKUSRPRF
> Work with User Profiles. This command allows you to work with a list of user profiles.

CRTUSRPRF
> Create User Profile. This command is used to identify new users to the system. You can specify that user's output queue and printer device.
>
> **Note:** You need special authority to create and delete user profiles.

DSPUSRPRF
> Display User Profile. This command allows you to display your user profile. It is useful for checking the user profile values. You can see the output queue and printer device values. To display user profiles of other users, you must have special authority.

CHGUSRPRF
> Change User Profile. This command allows you to change the current values of your user profile. If you have the correct level of authority, you can change the user profiles of other users.

DLTUSRPRF
> Delete User Profile. This command allows you to delete a user profile.
>
> **Note:** You need special authority to create and delete user profiles.

Commands Used with a Job Description

WRKJOBD
> Work with Job Descriptions. This command allows you change the attributes of a user's job description. To change the values of a job description other than your own requires special authority.

CRTJOBD
> Create Job Description. This command allows you to create job descriptions. When a user signs on, the values in the job description determine which output queue and printer device will be used.

CHGJOBD
> Change Job Description. This command allows you to change the values of a job description. If you have the correct level of authority, you can change other users' job description values.

Commands Used with Spooled Files

The following commands may be used to work with spooled files.

WRKSPLF

Work with Spooled Files. Allows you to display or print a list of all spooled files on the system.

WRKSPLFA

Work with Spooled File Attributes. Shows the current attributes of a spooled file.

CHGSPLFA

Change Spool File Attributes. Allows you to change some attributes of a spooled file, such as the output queue name or the number of copies requested, while the spooled file is on an output queue.

CPYSPLF

Copy Spooled File. Copies a spooled file to a specified database file. The database file may then be used for other applications, such as those using microfiche or data communications.

DLTSPLF

Delete Spooled File. Deletes a spooled file from the output queue.

DSPSPLF

Display Spooled File. Allows you to display data records of a spooled file.

HLDSPLF

Hold Spooled File. Stops the processing of an output file by a printer writer. The next spooled file in line is processed.

RLSSPLF

Release Spooled File. Releases a previously held spooled output file for processing by the printer writer.

SNDNETSPLF

Send Network Spooled Files. Allows you to send spooled files to another system.

Commands Used with Output Queues

The following commands may be used to create and control output queues.

WRKOUTQ

Work with Output Queue. Shows the overall status of all output queues, or the detailed status of a specific output queue.

WRKOUTQD

Work with Output Queue Description. Shows descriptive information for an output queue.

CLROUTQ

Clear Output Queue. Removes all spooled files from an output queue.

CRTOUTQ

Create Output Queue. Allows you to create a new output queue.

DLTOUTQ

Delete Output Queue. Deletes an output queue from the system.

HLDOUTQ

Hold Output Queue. Prevents all spooled files from being processed by the printer writer.

RLSOUTQ

Release Output Queue. Releases a previously held output queue for processing by the printer writer.

Commands Used with Writers

The following commands may be used to control writers.

WRKWTR

Work with Writers. Displays all the printers configured to the system by specifying WTR(*ALL) and the output queue (OUTQ) parameter.

STRPRTWTR

Start Printer Writer. Starts a printer writer to a specified printer device to print spooled files on that device.

STRRMTWTR

Start Remote Writer. Starts a remote writer to a specified output queue. The spooled files in that queue are then sent to a specified remote system for printing.

CHGWTR

Change Writer. Allows you to change some printer writer attributes, such as form type, number of file separator pages, or output queue attributes.

Remote writers allow only form type changes.

HLDWTR

Hold Writer. Stops the printer writer at the end of a record, at the end of a file, or at the end of a page.

RLSWTR

Release Writer. Releases a previously held printer writer for additional processing.

ENDWTR

End Writer. Ends a writer. For printer writers, makes the associated printer device available to the system.

Appendix C. Printer File Return Codes

This appendix contains descriptions of all major and minor return codes for printer files. These return codes are set in the I/O feedback area of the printer file. Return codes report the results of each operation. The appropriate return code is available to the application program that issued the operation. The program then checks the return code and acts appropriately. Refer to your high-level language manual for information about how to access these return codes.

The return code is a four-digit value: the first two digits contain the major code, and the last two digits contain the minor code. With some return codes, a message is also sent to the job log or the system operator message queue (QSYSOPR). You can refer to the message for additional information. Message IDs followed by an asterisk (*) may be received by applications while spooling output.

Note: Return codes that refer to a condition on the printer are available to the application program only when printing with SPOOL = *NO specified in the printer file. When SPOOL = *YES has been specified, the printer writer program is the program communicating with the printer, not your application program.

Major Code 00

> **Major Code 00** – Operation completed successfully.
>
> **Description:** The operation issued by your program completed successfully.
>
> **Action:** Continue with the next operation.

Code **Description/Action**

0000 **Description:** For output operations performed by your program, 0000 indicates that the last output operation completed successfully.

The notify messages are used after certain error conditions to give the operator the choice of continuing or canceling the printing of that file. If the reply is CANCEL, another message is issued with a nonzero return code.

Action: Your program may continue. One of the following diagnostic messages may have been issued to warn of an unusual condition that may be significant to your program even though it is not an error.

Messages:

CPA4001 (Inquiry)
        CPA5341 (Inquiry)

CPA4003 (Inquiry)
        CPA5342 (Inquiry)

CPA4004 (Inquiry)
        CPA5343 (Inquiry)

CPA4005 (Inquiry)
        CPA5344 (Inquiry)

CPA4007 (Inquiry)
> CPA5347 (Inquiry)

CPA4008 (Inquiry)
> CPA5348 (Inquiry)

CPA4009 (Inquiry)
> CPD4005 (Diagnostic)

CPA4010 (Inquiry)
> CPD4006 (Diagnostic)

CPA4011 (Inquiry)
> CPD4007 (Diagnostic)

CPA4012 (Inquiry)
> CPD4008 (Diagnostic)

CPA4013 (Inquiry)
> CPD4069 (Diagnostic)

CPA4014 (Inquiry)
> CPD4071 (Diagnostic)*

CPA4015 (Inquiry)
> CPD4072 (Diagnostic)

CPA4017 (Inquiry)
> CPF4032 (Diagnostic)

CPA4019 (Inquiry)
> CPF4033 (Diagnostic)

CPA4037 (Inquiry)
> CPF4056 (Diagnostic)

CPA4038 (Inquiry)
> CPF4057 (Diagnostic)

CPA4039 (Inquiry)
> CPF4239 (Escape)

CPA4040 (Inquiry)
> CPF4245 (Escape)

CPA4042 (Inquiry)
> CPF4249 (Escape)

CPA4043 (Inquiry)
> CPF4260 (Escape)*

CPA4046 (Inquiry)
> CPF4420 (Diagnostic)

CPA4047 (Inquiry)
> CPF4421 (Diagnostic)

CPA4048 (Inquiry)
> CPF4905 (Notify)*

CPA4065 (Inquiry)
> CPF4913 (Diagnostic)

CPA4066 (Inquiry)
> CPF4914 (Diagnostic)

```
        CPA4072 (Inquiry)*
            CPF4916 (Notify)*

        CPA4073 (Inquiry)
            CPF4918 (Notify)*

        CPA4074 (Inquiry)
            CPF4919 (Notify)*

        CPA4075 (Inquiry)
            CPI4015 (Informational)

        CPA4076 (Inquiry)
            CPI4016 (Informational)

        CPA4251 (Inquiry)
            CPI4017 (Informational)

        CPA4256 (Inquiry)
            CPI4018 (Informational)

        CPA5335 (Inquiry)
            CPI4019 (Informational)

        CPA5339 (Inquiry)
            CPI4020 (Informational)

        CPA5340 (Inquiry)
            CPI4024 (Informational)
```

Major Code 80

> **Major Code 80** – Permanent system or file error (nonrecoverable).
>
> **Description:** A nonrecoverable file or system error occurred. Recovery is unlikely until the problem causing the error has been corrected.
>
> **Action:** The following general actions can be taken by your program for each 80xx return code. Other specific actions are given in each return code description.
> - Continue processing without the printer.
> - Close the printer file and open the file again.
> - End.

Code   Description/Action

8081   **Description:** The operation was not successful because a system error condition was detected.

Action: Your printer may need to be varied off and then on again. Your program can either:
- Continue processing without the printer.
- Close the device file and open the file again.
- End.

Messages:

CPF4182 (Escape)*
        CPF5409 (Escape)

CPF4289 (Escape)
        CPF5410 (Escape)

CPF4510 (Escape)*
      CPF5414 (Escape)

CPF4516 (Escape)
      CPF5416 (Escape)

CPF4552 (Escape)
      CPF5418 (Escape)

CPF4591 (Escape)
      CPF5423 (Escape)

CPF5159 (Escape)
      CPF5429 (Escape)

CPF5196 (Escape)
      CPF5431 (Escape)*

CPF5246 (Escape)
      CPF5433 (Escape)

CPF5257 (Escape)*
      CPF5434 (Escape)

CPF5261 (Escape)
      CPF5447 (Escape)

CPF5262 (Escape)*
      CPF5453 (Escape)

CPF5401 (Escape)
      CPF5507 (Escape)

CPF5408 (Escape)

8082   **Description:** The operation attempted was not successful because the printer is unusable. This may occur because:

- A cancel reply has been taken to an error recovery message for the device.
- A cancel reply was returned to a maximum records reached inquiry message.
- The printer has been held by a Hold Communications Device (HLDCMNDEV) command.

No operations should be issued to the device.

Action: Communications with the printer cannot be resumed until the device has been reset to a varied-on state. If the device has been held, use the Release Communications Device (RLSCMNDEV) command to reset the device. If the device is in an error state, vary the device off and then on again. Once the device is reset, normal operation can be started again by reopening the printer file.

Messages:
    CPF4502 (Escape)
    CPF5104 (Escape)
    CPF5116 (Escape)*
    CPF5269 (Escape)

80B3   **Description**: The open operation was not successful because the printer file or printer device is not available.

Action: The printer file cannot be opened again until the necessary resources are available. Your program can wait for the resources to become available, then issue another open operation. Otherwise, you may continue other processing or end the program. The Work with Configuration Status (WRKCFGSTS) command may be used to determine whether the printer device is in use or not varied on. If the device is in use, the WRKCFGSTS command will also identify the job that is using it.

Consider increasing the WAITFILE parameter with the Change Printer File (CHGPRTF) or Override with Printer File (OVRPRTF) command to allow more time for the file resources to become available.

Messages:

```
CPF4128 (Escape)*
CPF9808 (Diagnostic)*
```

80C0 **Description**: A nonrecoverable error has occurred on the printer device.

Action: Your printer may need to be varied off and then on again. Your program can either:

- Continue processing without the printer.
- Close the printer file and open the file again.
- End.

Messages:

CPF4262 (Escape)
      CPF5413 (Escape)

CPF4509 (Escape)
      CPF5419 (Escape)

CPF5103 (Escape)
      CPF5420 (Escape)

CPF5247 (Escape)
      CPF5430 (Escape)

CPF5412 (Escape)
      CPF5437 (Escape)

80EB **Description**: An open operation was not successful because an open option that was not valid or an invalid combination of options was specified in your program, in the printer file, or in an override command.

Action: Close the printer file, correct the problem, and issue the open operation again. See the individual messages to determine what options are not valid.

Messages:

CPD4012 (Diagnostic)
      CPF4209 (Escape)

CPD4013 (Diagnostic)*
      CPF4214 (Escape)*

CPD4020 (Diagnostic)
      CPF4217 (Escape)

CPD4021 (Diagnostic)*
      CPF4219 (Escape)

CPD4023 (Diagnostic)
        CPF4224 (Escape)

CPD4024 (Diagnostic)
        CPF4237 (Escape)*

CPD4025 (Diagnostic)
        CPF4238 (Escape)

CPD4033 (Diagnostic)
        CPF4263 (Escape)*

CPF411E (Escape)
        CPF4264 (Escape)*

CPD4034 (Diagnostic)*
        CPF4295 (Escape)*

CPD4036 (Diagnostic)*
        CPF4296 (Escape)*

CPD4037 (Diagnostic)*
        CPF4335 (Escape)

CPD4038 (Diagnostic)*
        CPF4336 (Escape)

CPF4133 (Escape)
        CPF4337 (Escape)

CPF4138 (Escape)*
        CPF4338 (Escape)

CPF4139 (Escape)*
        CPF4339 (Escape)*

CPF4148 (Escape)
        CPF4340 (Escape)

CPF4156 (Escape)
        CPF4345 (Escape)

CPF4157 (Escape)*
        CPF4352 (Escape)

CPF4159 (Escape)*
        CPF4637 (Escape)

CPF4162 (Escape)
        CPF5370 (Escape)

CPF4181 (Escape)*

CPF4196 (Escape)*

CPF4206 (Escape)*

80ED   **Description**: An open operation was not successful because the record
        format descriptions in the printer file have changed since your program
        was compiled.

        Action: Close the printer file and end the program. Determine whether the
        changes affect your application program. If they do, then recompile the
        program. If the changes do not affect your program, the file should be
        changed or overridden to LVLCHK(*NO). When LVLCHK(*NO) is
        specified, the system does not compare the record format descriptions.

Messages:

```
CPF4131 (Escape)*
```

80EF **Description**: An open operation was not successful because your program is not authorized to the printer device.

Action: Close the file, correct the problem, then issue the open operation again. Obtain authority to the device from your security officer or the device owner.

Messages:

```
CPF4104 (Escape)*
```

80F8 **Description:** An operation was not successful because the file is marked in error.

Action: Close the file. Refer to messages in the job log to determine what errors occurred. Take the appropriate recovery action for those errors.

Messages:

```
CPF4132 (Escape)*
CPF5129 (Escape)*
CPF5293 (Escape)*
CPF5427 (Escape)*
```

Major Code 81

Major Code 81 – Permanent device error (nonrecoverable).

Description: A nonrecoverable device-related error occurred during an I/O operation. Any attempt to continue using this printer device will probably fail again until the cause of the problem is found and corrected.

Action: The following general actions can be taken for each 81xx return code. Other specific actions are given in each return code description.

- Continue processing without the printer device.
- Close the file, correct the problem, and open the file again. If the operation is still unsuccessful, try it again only a limited number of times. (The number of times should be specified in your program.)
- End.

Several return codes indicate that an error condition must be corrected by varying the device off and on again.

Code **Description/Action**

8181 **Description**: A system error condition was detected during an I/O operation to the printer device.

Action: Close the file. You may need to vary the device off and on again to clear the error. Determine the cause of the failure from the accompanying message. Check for any system operator messages indicating that additional corrective action must be performed. Open the file again to continue.

Messages:

```
CPF4289 (Escape)
CPF4552 (Escape)
```

```
CPF4553 (Escape)
CPF5105 (Escape)
CPF5159 (Escape)
CPF5507 (Escape)
```

8191   **Description:** The operation was not successful because a permanent line error occurred, and the system operator took a recovery option in response to the line error message. (You can find out what type of line error occurred by asking the system operator.) The device has been marked unusable.

Action: Close the file. Vary the device off and on again to clear the error. Open the file again to continue.

Messages:
```
CPF4146 (Escape)
CPF4193 (Escape)
CPF4526 (Escape)
CPF4542 (Escape)
CPF5128 (Escape)
CPF5198 (Escape)
```

8197   **Description:** A nonrecoverable error condition was detected at the device.

Action: Close the file. Vary the device off and on again to clear the error. Refer to the accompanying error message for additional information regarding the source of the specific error detected. Open the file again to continue.

Messages:

CPF4149 (Escape)
```
       CPF4583 (Escape)
```
CPF4192 (Escape)
```
       CPF5106 (Escape)
```
CPF4197 (Escape)
```
       CPF5143 (Escape)
```
CPF4216 (Escape)
```
       CPF5199 (Escape)
```
CPF4524 (Escape)
```
       CPF5201 (Escape)
```
CPF4533 (Escape)
```
       CPF5268 (Escape)
```
CPF4538 (Escape)
```
       CPF5360 (Escape)
```

81C2   **Description:** The operation issued by your program was not successful because the Systems Network Architecture (SNA) session with the printer is not active.

Action: Close the file. Vary the device off and on again to clear the error. Open the file again to continue.

Messages:
```
CPF5422 (Escape)
```

Major Code 82

Major Code 82 – Open operation failed.

Description: An attempt to open the printer file was not successful. The error may be recoverable or permanent, but is limited to the printer device. Recovery is unlikely until the problem causing the error has been corrected.

Action: The following general actions can be taken for each 82xx return code. Other specific actions are given in each return code description. You can either:

- Continue processing without the device.
- Close the file, correct the problem, and open the file again. A subsequent operation could be successful if the error occurred because of some temporary condition such as the device being in use at the time.

  If the operation is still unsuccessful, try it again only a limited number of times. (The number of times should be specified in your program.)
- End.

Several return codes indicate that an error condition must be corrected by changing a value in the file. To change a parameter value for the file, use the Change Printer File (CHGPRTF) or the Override with Printer File (OVRPRTF) command.

Code   Description/Action

8281 **Description:** A system error condition was detected on an open operation that was not successful. The printer file may previously have been in error, or the printer file could not be opened due to a system error.

Action: Your printer may need to be varied off and then on again to clear the error. Your program can either:

- Continue processing without the printer.
- Close the file, correct the problem, and open the file again.
- End.

Determine the cause of the failure from the accompanying message.

Messages:

```
CPF4168 (Escape)*
```

8282 **Description:** The open operation was not successful because the printer device is unusable. This may occur because a cancel reply has been taken to an error recovery message for the printer or because the printer has been held by a Hold Communications Device (HLDCMNDEV) command. No operations should be issued to the device.

Action: Close the file. Communications with the printer cannot be resumed until the device has been reset to a varied-on state. If the device has been held, use the Release Communications Device (RLSCMNDEV) command to reset the device. If the device is in an error state, vary the device off and then on again. Once the device is reset, normal operation can be started by opening the printer device file again.

Messages:

```
CPF4110 (Escape)
CPF4298 (Escape)
CPF4354 (Escape)
```

8291 **Description:** A permanent line error occurred on an open operation. The printer device has been marked unusable.

Action: Close the file. Vary the device off and on again to clear the error. Open the file again to continue.

Messages:

    CPF4179 (Escape)
    CPF4291 (Escape)

82A6 **Description:** The open operation failed because of a Systems Network Architecture (SNA) protocol violation.

Action: Ensure that the printer with which your program is communicating is configured properly. Refer to the device response codes in the accompanying error message for additional information regarding the specific error detected.

Messages:

CPF4124 (Escape)
        CPF4533 (Escape)

CPF4190 (Escape)
        CPF5103 (Escape)

CPF4192 (Escape)
        CPF5143 (Escape)

CPF4527 (Escape)
        CPF5453 (Escape)

82AA **Description:** The open operation was not successful because the printer device description was not found.

Action: Your program can continue without the printer, attempt to use a different printer, or end.

Verify that the name of the printer was correctly specified in the DEV parameter on the CRTPRTF, CHGPRTF, OVRPRTF, or CRTPRTF command.

Messages:

    CPF4103 (Escape)*

82B3 **Description:** The open operation was not successful because the printer you requested is in use in another file in your job.

Action: Close both of the printer device files, then open the one that you want to use again.

Messages:

    CPF4106 (Escape)

82EE **Description:** An open operation was attempted to a device that is not supported for a printer file.

Your program is attempting to open a device that is not a valid printer.

Action: Your program can continue without the printer, attempt to use a different printer, or close the file and end.

Verify that the name of the printer was specified correctly on the CHGPRTF or OVRPRTF command.

Messages:

```
                          CPF4105 (Escape)
```

82EF   **Description:** An open operation was attempted for a device that the user is not authorized to, or that is in service mode.

Action: Your program can continue without the printer, attempt to use a different printer, or end.

Close the file, correct the problem, and then issue the open operation again.

For authority errors, obtain authority to the device from your security officer or device owner. If the device is in service mode, the system service tools (SST) function is currently using the device. Wait until the device is available to issue the operation again.

Messages:

```
    CPF4104 (Escape)*
    CPF4186 (Escape)
    CPF9802 (Diagnostic)*
```

Major Code 83

Major Code 83 – Device error occurred (recoverable).

Description: An error occurred during an I/O operation, but the printer device is still usable. Recovery within your program might be possible.

Action: The following general actions can be taken for each 83xx return code. Other specific actions are given in each return code description.

- Continue processing without the printer device.
- Correct the problem and continue processing with the printer device. If the attempt to recover from the operation is unsuccessful, try it again only a limited number of times. (The number of times should be specified in your program.)
- End.

Several return codes indicate that an error condition must be corrected by changing a value in the file. To change a parameter value for the file, use the Change Printer File (CHGPRTF) or Override with Printer File (OVRPRTF) command.

Code   **Description/Action**

8319   **Description:** A negative response was received to the last printer operation attempted by your program. The error may have been caused by the user pressing the Cancel key on the printer.

Action: Your program can try a different operation, or close the file and end. Refer to the device response code in the accompanying message to determine why the operation was rejected. Correct the error in your program before attempting to try the operation again.

Messages:

```
    CPF4158 (Escape)
    CPF4531 (Escape)
    CPF5050 (Escape)
```

831D   **Description:** The operation just attempted by your program was rejected because a parameter was not valid, out of limits, or missing.

Action: Your program can bypass the failing step and continue, or close the file and end. Refer to the accompanying message to determine what parameter was incorrect. Correct the error in your program before attempting to try the operation again.

Messages:

CPD4016 (Diagnostic)*
       CPF5275 (Escape)

CPD4017 (Diagnostic)*
       CPF5276 (Escape)*

CPD4027 (Diagnostic)*
       CPF5288 (Escape)*

CPD4028 (Diagnostic)*
       CPF5289 (Escape)*

CPD4029 (Diagnostic)*
       CPF5324 (Escape)*

CPD4030 (Diagnostic)*
       CPF5359 (Escape)*

CPD4041 (Diagnostic)*
       CPF5363 (Escape)

CPF4909 (Notify)*
       CPF5366 (Escape)*

CPF5108 (Escape)*
       CPF5367 (Escape)*

CPF5148 (Escape)*
       CPF5368 (Escape)*

CPF5273 (Escape)*

831E   **Description:** The operation just issued by your program was not valid or an invalid combination of operations was specified.

      Action: Your program can bypass the invalid operation and continue, or close the file and end. Refer to the accompanying message to determine why the operation was rejected. Correct the error in your program before attempting to try the failing operation again.

      Messages:

CPD4015 (Diagnostic)*
       CPF5290 (Escape)*

CPD4018 (Diagnostic)*
       CPF5320 (Escape)*

CPD4031 (Diagnostic)
       CPF5321 (Escape)*

CPF4915 (Notify)*
       CPF5322 (Escape)*

CPF5149 (Escape)*
       CPF5323 (Escape)*

CPF5185 (Escape)*
       CPF5325 (Escape)*

```
CPF5245 (Escape)*
       CPF5362 (Escape)*
```

831F **Description:** A length that is not valid was specified on the operation.

On an output operation, your program has tried to send a data record having a length that exceeds the maximum record length allowed for the printer or the page size defined for the file. If you are using direct I/O, you have exceeded the maximum number of bytes allowed per page. The data has been truncated.

Action: Issue the output operation again with a smaller output length. The record length for a program-described printer file cannot exceed the page size. The record length for any printer file must be no greater than 32 767 characters.

Messages:
```
CPF4906 (Notify)*
CPF5160 (Escape)
```

8343 **Description:** The designated page overflow line number has been reached.

Action: Your program should take whatever application dependent action is appropriate. This may include printing page totals or a running foot line.

Messages:
```
CPF5004 (Status)*
```

83E0 **Description:** Your program attempted to issue an operation using a record format that was not defined for the printer file, or omitted the record format name.

Action: Check the name of the record format in your program to be sure it is correct. Then check that the record format is defined properly in the DDS for the file.

Messages:
```
CPF5186 (Escape)*
CPF5187 (Escape)*
```

83F6 **Description:** Your program sent invalid data to the printer. The data type may be incorrect for the field in which it is used.

Action: Check the name of the record format in your program to be sure it is correct. Verify that the data definition statements in your program match the output record defined in the DDS for the file. Correct the error in your program before attempting to repeat the failing operation.

Messages:

```
CPD4014 (Diagnostic)*
       CPF5075 (Notify)*

CPD4022 (Diagnostic)*
       CPF5234 (Escape)*

CPD4026 (Diagnostic)*
       CPF5246 (Escape)

CPD4035 (Diagnostic)*
       CPF5261 (Escape)

CPD4516 (Informational)
       CPF5297 (Escape)*
```

CPD4591 (Escape)
     CPF5364 (Escape)

CPF4634 (Escape)
     CPF5365 (Escape)

CPF4635 (Escape)
     CPF5369 (Escape)

CPF4636 (Escape)
     CPF5372 (Escape)

CPF4642 (Escape)
     CPF5373 (Escape)

CPF4643 (Escape)
     CPF5374 (Escape)

CPF4644 (Escape)
     CPF5375 (Escape)

CPF4645 (Escape)
     CPF5376 (Escape)

CPF4646 (Escape)
     CPF5377 (Escape)

CPF4647 (Escape)
     CPF5411 (Escape)

Appendix D. Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts

Understanding the relationship between these elements on the iSeries server gives you flexibility in preparing your documents for printing.

The first part of this appendix discusses these elements and the second part contains tables that show how the elements work together.

Fonts and the iSeries server

The iSeries server comes with a certain variety of fonts called the IBM-supplied compatibility fonts or the compatibility set. These fonts provide a range of font styles that support different types of printers that can be attached to the system.

The iSeries server also comes with the IBM WorldType fonts. These are Unicode fonts that are provided in the TrueType/OpenType format. The WorldType fonts contain characters to support language and scripts from around the world. Presently, over 52,000 glyphs are contained in a single typeface. Subsets of this data are also available to allow a smaller file size to support certain geographic areas. See "WorldType fonts" on page 378 for details.

The compatibility set can be supplemented by installing IBM licensed programs that provide additional fonts, creating your own fonts on the iSeries server, or purchasing them from other companies.

See the "Bibliography" on page 543 for a list of manuals that contain additional information about fonts and font-related products.

Downloading

Downloading is the process of sending something (for example, character sets or code pages) to another computer or printer.

Some printers do not have fonts built into them. The host system can send (download) character sets and code pages to the printer with the document or ahead of time, to be stored for future use.

Considerations When Using 240- and 300-Pel Fonts

Most printers support 240-pel font character sets. However, the 4028, and 3935 printers support only 300-pel font character sets. If printed output is sent to a combination of 4028 and other IPDS printers by applications that use font character sets that are downloaded from the iSeries server, some special considerations apply:

- Font name

  The font character set used with the document must exist in both the 240- and 300-pel versions and have the same name. The application selects the font character set by name, not by pel density.

- Font character set selection

  The system knows which pel density a printer supports and downloads the correct font character set to the printer.

- Libraries

377

Font character sets are font resource objects regardless of pel density. The 240-pel and 300-pel fonts must be stored in separate libraries.

When an application program calls for a particular element (font or code page), the system looks for it in the printer being used or in the iSeries server. If it cannot find the designated font or code page, a substitution or mapping takes place. Information concerning the substitution of fonts and code pages is contained in this appendix.

If you need to know more about fonts, font character sets, code pages, or coded fonts, read on. If not, you can skip to "Font Substitution Tables" on page 391 for information describing how fonts are substituted between printers and computers.

WorldType fonts

The WorldType fonts are IBM's strategic Unicode fonts. They contain characters to support language and scripts from around the world. Presently, over 52,000 glyphs are contained in a single typeface. Subsets of this data are also available to allow a smaller file size to support certain geographic areas.

The WorldType fonts are supplied to the iSeries in option 43 ("Additional Fonts") of OS/400. They are provided as stream files in the TrueType/OpenType format. When this option is installed, the font files are stored in the following integrated file system directory:

\QIBM\ProdData\OS400\Fonts\TTFonts

You must use the FONTNAME DDS keyword to select the WorldType fonts. Unlike the other supported font resources, the WorldType fonts are referred to not by the file or object name, but by their *full font name*.

The following font names are supplied with the iSeries:
- Monotype Sans WT
- Monotype Sans WT J
- Monotype Sans WT K
- Monotype Sans WT SC
- Monotype Sans WT TC
- Monotype Sans Duospace WT
- Monotype Sans Duospace WT J
- Monotype Sans Duospace WT K
- Monotype Sans Duospace WT SC
- Monotype Sans Duospace WT TC
- Times New Roman WT
- Times New Roman WT J
- Times New Roman WT K
- Times New Roman WT SC
- Times New Roman WT TC
- Thorndale Mono WT
- Thorndale Mono WT J
- Thorndale Mono WT K
- Thorndale Mono WT SC
- Thorndale Mono WT TC

WorldType fonts can be used only with printer files with a device type of *AFPDS.
Currently, they can be printed only with the host print transform function to
image-capable ASCII printers.

Font Character Sets and Font Global Identifiers (FGID)

Fonts are a family or assortment of characters. Three elements usually provide a
font identity:

- Type Family

  Courier is an example of a type family.

- Typeface

  Style, weight (for example, italic or bold), and width (normal or expanded)
  define typeface.

  Normal means the usual size of characters, while expanded means that the
  character is wider than normal.

- Type Size

  Fonts can range from small (4 point) to large (72 point).

For example, a font could be identified as:

Type family
       Sonoran Serif

Typeface
       Roman medium normal

Type Size
       10-point

Font Character Sets

Fonts are named in a number of ways. One way is with a character set name.
These character sets are downloaded to the printer. Multiple code pages can be
used with a single character set. For valid code pages that can be used with a
character set, see the manual *About Type: IBM's Technical Reference for 240-Pel
Digitized Type* , GS544-3516.

Some font character sets come with the iSeries server; some can be downloaded
from a System/390 to an iSeries server; some can be received from another iSeries
server; and some are available as licensed programs.

The following printers accept downloaded font character sets:

- 3112 (has resident fonts also)
- 3116 (has resident fonts also)
- 3130 (has resident fonts also)
- 3160 (has resident fonts also)
- 3812 (has resident fonts also)
- 3816 (has resident fonts also)
- 3820
- 3825
- 3827
- 3828 (MICR printer)
- 3829

- 3831
- 3835
- 3900–001
- 3900–AFCCU (has resident fonts also)
- InfoPrint 3000 (has resident fonts also)
- InfoPrint 4000 (has resident fonts also)
- 3912 (has resident fonts also)
- 3916 (has resident fonts also)
- 3930 (has resident fonts also)
- 3935 (has resident fonts also)
- 4028 (has resident fonts also)
- 4312 (has resident fonts also)
- 4317 (has resident fonts also)
- 4324 (has resident fonts also)
- InfoPrint 20 (has resident fonts also)
- InfoPrint 32 (has resident fonts also)

Note: Any printer attached to Print Services Facility for OS/2 (PSF for OS/2) accepts downloaded fonts.

The use of font character sets provides consistent or similar fonts across printers. For example, a document created at one location using a specific font character set could be sent to a different location, printed on a different model printer, and still look the same.

With some exceptions, the above printers support font character sets that are 240 pels. The InfoPrint 3000, InfoPrint 4000, InfoPrint 20, InfoPrint 32, 4028, 3130, 3935, 4312, 4317, and 4324 printers support 300-pel fonts. A pel is a picture element, representing the number of dots in a square inch (for example, 240 across and 240 down).

The 3130, InfoPrint 3000, and InfoPrint 4000 printers support both 240 and 300 pel fonts. The operator can select which mode the printer is in through the printer operator panel.

Naming Convention for Font Character Sets
Font character set names on OS/400 can be up to 8 characters long. Each character or group of characters tells something about the font character set.

For example, in the font character set name C0D0GT10:

C0   The **C0** means that this object is a font character set.

D   The **D** indicates the origin of the font. In this example, C0D0GT10 is a font character set designed for Document Control Facility (DCF) for a 3800 Model 1 printer or a 3825 printer.

0   This **0** indicates that this font is for uniformly spaced and mixed-pitch font character sets.

GT10   The **GT10** indicates the type family, typeface, and pitch for uniformly spaced and mixed-pitch fonts. In this example, the GT10 means that this font character set is a Gothic Text style and the characters are 10 pitch or 10 characters per inch.

For more information about font character sets, see the manual *About Type: IBM's Technical Reference for 240-Pel Digitized Type* , GS544-3516.

Selecting Font Character Sets

Selecting a font character set to use with an application program is done by specifying the 8-character font character set name as the value on the FNTCHRSET parameter of the printer file.

If you choose to use font character sets with your applications, you must also specify a code page (by providing a value for the CDEPAG parameter of the printer file being used).

Substituting Font Character Set

Substitution is determined by OS/400, based on which font character sets are specified in the application, the type of printer to be used, and the value assigned to the fidelity parameter of the printer file being used (*CONTENT or *ABSOLUTE).

Example 1: Assume:
- The application calls for font character set C0D0GB10 (Gothic Bold, 10 pitch).
- The printer supports only resident fonts.
- The fidelity parameter value is *CONTENT.

In this example, the spooled file will print, with substituted font ID 39 (Gothic Bold 10 pitch) because the fidelity parameter value is *CONTENT. If the fidelity parameter value had been *ABSOLUTE, the spooled file would be held on the output queue and it would not print.

Example 2: Assume:
- The application calls for FGID 51 (Matrix Gothic).
- The printer supports only downloaded font character sets.
- The fidelity parameter value is *CONTENT.

In this example, the spooled file will print. OS/400 substitutes a font character set (C0S0CR10, Courier Roman 10 pitch) for FGID 51. This is not an exact match. The system matched (as closely as possible) the font character to the FGID specified in the application.

Note: In this example, if the fidelity parameter was *ABSOLUTE, the spooled file would be HELD.

Font Global Identifiers (FGIDs)

Another method of naming a font is by a font global identifier (FGID). An FGID names a type family and a typeface.

FGIDs are identified by a number such as 3, 8, or 11.

There is a different FGID assigned for the same type family but different typeface. For example, a Courier Roman Medium 10 pitch (characters-per-inch) is FGID 11 and Courier Roman Bold 10 pitch (characters per inch) is FGID 46.

Below is an example of FGID 11. The text in the box represents how data would print if your application uses FGID 11.

```
    FGID 11 is a monospaced courier font that
    will print 10 characters per inch.
```

Printers with resident fonts use FGIDs to name the resident fonts. Depending on the technology used with the printer, resident fonts can be stored on font cards, diskettes, in the memory of the printer, or mechanically on a font element or daisy wheel.

The following printers have resident fonts:
- 3112 (can also accept downloaded fonts)
- 3116 (can also accept downloaded fonts)
- 3130 (can also accept downloaded fonts)
- 3160 (can also accept downloaded fonts)
- 3812 (can also accept downloaded fonts)
- 3816 (can also accept downloaded fonts)
- 3930
- 3912, 3916, or 4028 (can also accept downloaded fonts)
- 3935 (can also accept downloaded fonts)
- 4214
- 4224
- 4230
- 4234 Models 8 and 12
- 4247
- 4312 (can also accept downloaded fonts)
- 4317 (can also accept downloaded fonts)
- 4324 (can also accept downloaded fonts)
- 5219
- 5224
- 5225
- 6400
- 6408
- 6412
- 3900–AFCCU (can also accept downloaded fonts)
- InfoPrint 20 (can also accept downloaded fonts)
- InfoPrint 32 (can also accept downloaded fonts)
- InfoPrint 3000 (can also accept downloaded fonts)
- InfoPrint 4000 (can also accept downloaded fonts)

To find out which fonts are supported by a printer, check the reference manual for that printer.

Selecting Resident Fonts
Selecting a resident font to use with an application program is done by specifying an FGID value on the FONT parameter of the printer file.

Font Substitution

Substitution can be one FGID for another, an FGID for a font character set, or a font character set for an FGID.

Example 1: Assume:
- Your application calls for a font character set (FNTCHRSET specified on the printer file), for example, C0S0CR10 for Courier Roman medium 10 pitch.
- The printer is a 4224 and has resident fonts identified by FGIDs.
- FGID 11 will be substituted for C0S0CR10 and sent to the printer.

In this example, the system substitutes a font that is resident on that printer.

Example 2: Assume:
- Your application calls for a font (specified on the FONT parameter of the printer file). The font specified is font 26 (Gothic Matrix, Roman medium 10 pitch) and the printer is a 3812.
- You decide to print the document on a 4019 printer. Font 26 is not supported on the 4019.

In this example, the system substitutes font 11 (Courier, Roman medium 10 pitch)

You can figure out such substitutions by looking at Table 37 on page 404.

Example 3: Assume:
- Your application uses a font (specified on the FONT parameter of the printer file). The font specified is font 40 (Gothic, Roman medium 10 pitch).
- The printer you are going to print on supports only font character sets (for example, a 3827).

In this example, the system substitutes font character set C0D0GT10 (Gothic Text, Roman medium 10 pitch).

You can figure out such substitutions by looking at Table 40 on page 442.

Code Pages

Code pages come in two types:
- Code page (standalone)
- Character set and code page combination (referred to as a CHRID).

Code pages are groups of characters. Within a code page, there are unique hexadecimal identifiers assigned to each of the characters.

As you enter your text at a computer keyboard, each keyboard character is translated into a code point. When the text is printed, each code point is matched to a character ID on the code page you specified. The character ID is then matched to the image (raster pattern) of the character in the character set you specified.

Some of these characters can be repeated in different code pages and have a different hexadecimal identifier assigned to them. Conversely, the hexadecimal identifier can be the same, but the characters will be different. Therefore, if you have applications that use certain characters contained in only one particular code page, it is important to know which code page you are using.

Below is a diagram of two code pages: code page 37 and code page 285. They are printed in a 10-characters-per-inch (courier 10) font. Notice the different character that occurs at code point hex '5B'. One is a $ sign and the other an English pound or currency sign. This example shows that different characters will print depending on the code page you specify even though you are using an identical font style.

Code page 37 with courier 10   Code page 285 with courier 10

Code point 5B   Code point 5B

```
   0 1 2 3 4 5 6 7 8 9 A B C D E F        0 1 2 3 4 5 6 7 8 9 A B C D E F
4        â ä à á ã å ç ñ ¢ \ < ( + |   4        â ä à á ã å ç ñ $ \ < ( + |
5  & é ê ë è í î ï ì ß ! $ * ) ; ¬     5  & é ê ë è í î ï ì ß ! £ * ) ; ¬
6  - / Â Ä À Á Ã Å Ç Ñ ] , % _ > ?     6  - / Â Ä À Á Ã Å Ç Ñ ] , % _ > ?
7  ø É Ê Ë È Í Î Ï Ì ` : # @ ' = "     7  ø É Ê Ë È Í Î Ï Ì ` : # @ ' = "
8  Ø a b c d e f g h i « » ð ý þ ±     8  Ø a b c d e f g h i « » ð ý þ ±
9  º j k l m n o p q r ª o æ , Æ ¤     9  º j k l m n o p q r ª o æ , Æ ¤
A  µ ~ s t u v w x y z ¡ ¿ Ð Ý Þ ®     A  µ ~ s t u v w x y z ¡ ¿ Ð Ý Þ ®

E  \   S T U V W X Y Z ² Ô Ö Ò Ó Õ    E  \   S T U V W X Y Z ² Ô Ö Ò Ó Õ
F  0 1 2 3 4 5 6 7 8 9 ³ Û Ü Ù Ú       F  0 1 2 3 4 5 6 7 8 9 ³ Û Ü Ù Ú
```

RV2H330-0

Standalone Code Pages

Code pages supply consistent or similar characters across systems. For example, a document created at one location using a specific code page could be sent to a different location, printed on a different model printer, and still look the same.

Code pages must be downloaded to the printer for use.

The following printers accept downloaded code pages:
- 3112 (has resident fonts also)
- 3116 (has resident fonts also)
- 3130 (has resident fonts also)
- 3160 (has resident fonts also)
- 3812 (has resident fonts also)
- 3816 (has resident fonts also)
- 3820
- 3825
- 3827
- 3828 (MICR printer)
- 3829
- 3831
- 3835
- 3900–001
- 3900–AFCCU (has resident fonts also)
- InfoPrint 3000 (has resident fonts also)
- InfoPrint 4000 (has resident fonts also)
- 3900
- 3912 (has resident fonts also)
- 3916 (has resident fonts also)

- 3930 (has resident fonts also)
- 3935 (has resident fonts also)
- 4028 (has resident fonts also)
- 4312 (has resident fonts also)
- 4317 (has resident fonts also)
- 4324 (has resident fonts also)
- InfoPrint 20 (has resident fonts also)
- InfoPrint 32 (has resident fonts also)

Naming Convention for Code Pages

Like character sets, code pages are named in a number of ways. One way is with a code page name. These code pages are downloaded to the printer. The code page name can be up to 8 characters long. Code page names are used with character set names for printing on printers such as the 3820, 3825, 3827, or 3835.

Another way is with a code page global identifier (CPGID). CPGIDs are printer-resident code pages and have numbers for names (for example, 259 or 500). Generally, printers with resident fonts use CPGIDs to name the printer-resident code pages. CPGIDs are also used within CHRIDs.

For example, in the code page name T1V10500:

T   The **T** means that this object is a code page.

1   This always a 1.

V1   The **V1** means that this is version 1 of this code page.

0500 The **0500** is the code page name, number, or category. In this example 500 is the code page name.

Selecting Code Pages

Code pages are selected by specifying a certain value for the code page (CDEPAG) parameter of the printer file.

If you choose to use code pages with your applications, you must also specify a font character set (by providing a value for the FNTCHRSET parameter of the printer file being used).

Substituting Code Pages

Substitution of code pages occurs for the following reasons:
- The application specifies a code page that is resident on a printer and the printer being used does not have resident code pages.
- The application specifies a code page that is resident on the host system (iSeries server) and the printer being used has resident code pages (not capable of accepting downloaded code pages).
- The job requesting the code page is not authorized to it.
- The code page cannot be found.
- The job is not authorized to the library where the code page is stored.

Character Set and Code Page Combination (CHRIDs)

This type of code page is made up of a specific graphic character set and a specific code page and is referred to by a character identifier (CHRID).

These graphic character sets and code pages (CHRIDs) are used for fonts that are resident on the printer. They are used in conjunction with a font ID to obtain a resident font.

The following printers support CHRIDs:

- 3112
- 3116
- 3130
- 3160
- 3812
- 3816
- 3900–AFCCU
- 3912
- 3916
- 3930
- 3935
- 4028
- 4214
- 4224
- 4230
- 4234
- 4247
- 4312
- 4317
- 4324
- 5219
- 5224
- 5225
- 6400
- 6408
- 6412
- InfoPrint 20
- InfoPrint 32
- InfoPrint 3000
- InfoPrint 4000

Naming Convention for CHRIDs

The names of character identifiers (CHRIDs) are made up of two elements: graphic character set and code page. These two elements define a collection of characters. Below is an example of the multinational CHRID 697-500.

697   This is the name of the graphic character set.

Some graphic character sets identify a character set that is a subset of the code page. Others identify a character set that is equivalent to the code page.

500   This is the name of the code page.

Selecting CHRIDs

CHRIDs are selected by specifying a certain value for the character identifier (CHRID) parameter of the printer file. Additionally, a font ID value must be specified for the FONT parameter on the printer file.

Substituting CHRIDs

If the CHRID is not available on the printer your application is using, the system will substitute the CHRID that most closely matches the one requested by the application.

Coded Fonts

A coded font is the pairing of a font character set and a code page. Coded fonts allow users to specify a font character set and a code page with one value specified on the printer file.

Coded fonts available on the iSeries server can be viewed by using the Work with Font Resources (WRKFNTRSC) command.

Coded font names are read by the system and then translated to a font character set and a code page. These two elements are then sent to the printer.

Naming Convention for Coded Fonts

Unlike other uniformly spaced and mixed-pitch font components, coded font names are generally shortened by excluding the origin and reserved characters (the first two characters of their name). This is necessary because some Advanced Function Printing (AFP) licensed programs accept only 6 characters for coded font names. However, some applications can use coded fonts named with 6 or 8 characters.

Coded font names on the iSeries server are 6 or 8 characters long. Each character or group of characters tells something about the coded font.

For example, in the coded font name X0GT10:

X0   The X0 means that this object is a coded font.

XZ   The XZ means that this object is an outline coded font.

GT10   The **GT10** indicates the type family, typeface, and pitch for uniformly spaced and mixed-pitch fonts. In this example the GT10 means that this font character set is a Gothic Text style and the characters are 10 pitch or 10 characters per inch.

To find out which font character set and code page make up a coded font name, use the Work with Font Resources (WRKFNTRSC) command. This command allows you to specify the font resource to be worked with, the library it is in, and the attribute (coded font).

Additional naming conventions have been adopted to more explicitly name the code page used with a character set.

For more information about coded fonts, see the manual *About Type: IBM's Technical Reference for 240-Pel Digitized Type* , GS544-3516.

Selecting Coded Fonts

A coded font is selected by specifying the coded font name as the value on the coded font (CDEFNT) parameter of the printer file.

You can use the Work with Font Resources (WRKFNTRSC) command to view the coded fonts that are available on the system.

Substituting Coded Fonts

No substitution of coded fonts takes place on the iSeries server. If the coded font is not available, the document will not print.

You can use the MAPIGCFNT on the CRTPSFCFG and CHGPSFCFG commands to specify coded font names of the form X0nnnnnn should be mapped to XZnnnnnn. If the XZnnnnnn coded font is found it is used, if it is not found the X0nnnnnn coded font will be used.

Font Capturing

Font capturing refers to the ability of a printer to dynamically capture a host downloaded font. In previous releases, downloaded fonts were saved across jobs, but not across instances of the printer writer. With PSF/400 V4R2, the printer can capture a host downloaded font, if it supports the font capturing function. The captured font then appears like a printer-resident font, even if the printer has been powered off and back on. On the next instance of the printer writer, the need for subsequent font downloads is eliminated. The amount of data sent through the network is thereby reduced.

Captured fonts remain on the printer for an indefinite amount of time. The printer retains captured fonts until the space they occupy is needed for something else. The decision to discard captured fonts is made by the printer and is not under the control of the host printer writer.

Note: Even though a font may have been captured, the host font must still remain on the system because the printer may discard fonts to free up space, and the font may need to be reloaded. In addition, users must still have authority to the font object on the host in order to use it.

Activating Font Capturing

To activate font capturing, set the FNTCAPTURE parameter in the PSF configuration object to *YES. For more information about using the CRTPSFCFG command, see "Creating a PSF configuration object" on page 235. Any fonts eligible to be captured will then be sent to the printer with an indication that the printer may capture the font. It is important to note that the printer determines whether to capture a font based on available memory and free disk space.

Making Character Sets and Code Pages Eligible for Capturing

To make a font eligible to be captured, specify the name of a font character set or code page, and select FNTCAPTURE(*YES) in the Change Font Resource (CHGFNTRSC) command or the Create Font Resource (CRTFNTRSC) command. This will insert information into the font that tells the printer writer that this font is eligible for capture as well as a time and date stamp. The process of making a font eligible for capture is also called marking a font in some IBM operating environments.

Fonts supplied with the IBM AFP Font Collection (5648-113) are eligible to be captured as is. There is no need to use the CHGFNTRSC command to make them eligible.

Fonts that are created with the OS/2 Type Transformer, can be made eligible for capture by selecting the Capture check box in the Typeface Descriptions dialog box.

Eligibility Rules

Some fonts, especially very old fonts and custom built fonts, contain a value of zero for the Graphic Character Set Global Identifier (GCSGID) and Font Global Identifier (FGID) (both from the Font Descriptor (FND) structured field). Non-zero values for these fields is necessary to allow capturing. In these cases, the CRTFNTRSC command, or theCHGFNTRSC command will fail with a message indicating that the font was not marked as eligible for capture. These fonts will continue to be downloaded to the printer as usual.

Inline Fonts
Font resources that are sent inline with the print job are not eligible for capture and will always be downloaded.

Raster Fonts
For raster technology fonts, both the character set and the code page must be eligible for capturing for capturing to occur. If either is marked ineligible, or does not contain sufficient information to uniquely identify it, then the font will be downloaded.

Outline Fonts
In contrast with raster technology fonts, outline technology font character sets and code pages are captured independently of each other. For example, a code page could be captured, while a character set that is marked ineligible would be downloaded.

Migrating Font Libraries from Other Operating Environments

If you migrate a font library from MVS and the font library has been marked with the MVS utility APSRMARK, there is no need remark the fonts. The eligible or not eligible for capture attribute is analogous to the public or private designation in APSRMARK. Creating the font using the FNTCAPTURE(*FILE) on the CRTFNTRSC command will cause the font to be created with the information already contained in the font.

Considerations

Before using the font capturing function, make sure that you understand the considerations that are discussed in this section. Failure to do so may cause unpredicatable results, even for users of another system who may be sharing the printer.

How Captured Font Resources Are Identified
It is important to understand how font resources captured by the printer are distinguished from one another. The identity of a captured font resource is contained in the structured fields of the font resource itself. The source of this information is detailed below.

- Character sets

  The identity of a character set is formed by combining the Graphic Character Set Global Identifier (GCSGID), the Font Global Identifier (FGID) (both from the Font Descriptor (FND) structured field), and the time and date stamp (from the Begin Font (BFN) structured field).

- Code pages

The identity of a code page is formed by combining the Graphic Character Set Global Identifier (GCSGID), the Code Page Identifier (CPGID) (both from the Code Page Descriptor (CPD) structured field), and the time and date stamp (from the Begin Code Page (BCP) structured field.

When the configuration object has FNTCAPTURE(*YES), PSF uses this identifying information from the font to query the printer to see if it has a font with the same identifiers.

Because the identity of a font resource is carried within the font structured fields, it is imperative that the modified or customized fonts be remarked to distinguish them from the original font resource. Renaming a modified font, or placing it into a different library does not change the identity of a font as far as font capture is concerned. If a code page or character set that is marked eligible for capture is modified, the resource must be remarked by using the CHGFNTRSC command so that a new time and date stamp will be inserted into the resource. Failing to do so will result in the modified font resource still having the exact same identifiers as the original font resource. Having more than one font resource with the same identifying information makes it impossible to predict which one of the two resources will actually be used. This is especially important when a font resource exists in more than one library on the system, and when a font resource exists on multiple systems.

How Host Downloaded Resources are Identified
In contrast to captured fonts, host downloaded fonts remain in the printer across job boundaries, but not across instances of the printer writer. With host downloaded fonts, the printer writer knows the name and library of the font resource object and can insure that font resources with the same name but from different libraries are not mistaken for each other.

Guidelines
When using font capturing, make sure you know where the font resources are coming from. Consider the following suggestions to reduce the chance of unwanted font resources being captured by the printer.

1. Run the CHGFNTRSC command to update the time and date stamp for all modified font resources. If a modified font resource is not appropriate for use by everyone, including users from another system who share the printer, it should be marked as not eligible for capture.

2. Treat font resources, and the commands that operate on them as system resources to be used only by those users who need them. Allowing users to make personal copies of font resources increases the risk of having different fonts with the same identifying information.

3. To prevent font resources in user libraries from being captured, the PSF configuration object may be configured with USRRSCLIBL(*NONE). This causes the printer writer to ignore the user resource library list and search only those libraries in the resource library list.

4. If you are concerned about modified font resources in user libraries, or are sharing a printer with another system, you can take the following steps to assure that the writer does not use captured fonts. This may be appropriate for critical production jobs.

    a. Turn off the capturing function by setting FNTCAPTURE(*NO) in the PSF Configuration object. This prevents any fonts from being captured and also assures that no captured resources will be used in the printer writer.

b. Run the CHGFNTRSC command with FNTCAPTURE(*NO) against the font resources used by the production jobs. This assures that the font resource will be downloaded, and that a captured font resource is never used.

Sharing Printers among Systems

If you share a printer among systems, it is imperative that all systems control the font resources that are allowed to be captured by the printer. If user modified fonts are captured by the printer, they become available for use by any system that shares the printer. This may produce undesirable results, because the modified font resource from another system may not be what is intended for use.

Security

Do not mark sensitive fonts, such as signatures and MICR fonts, as eligible to be captured. This is because there is no means for the host to clear captured fonts from the printer, and it is possible that an unauthorized person could access the captured font, even from another system.

Font Substitution Tables

The following tables contain information on fonts, character identifiers and other printing characteristics.

Notes:

1. The iSeries server supports Font Object Content Architecture (FOCA) 2 font character sets. This means it does not support the font character sets used by the various models of the IBM 3800 printer. For more information on FOCA 2 fonts, see the *Font Object Content Architecture Reference* , S544-3285.
2. Font global identifier (FGID) is used interchangeably with font ID in these tables.

Font Attributes

Font attributes are characteristics about a font that combine to give a font identity. An example of a font attribute would be bold or italic.

Table 36 on page 395 is a list of OS/400-supported fonts and their attributes.

Font Substitution

Table 37 on page 404 contains font ID substitution information which identifies which fonts are supported by certain printers.

For example: if your application specified a certain font ID that your printer did not support, you could find out which printers support that font and route your printed output to a printer that supports that font ID. Also, this table provides information on what fonts are substituted if the font id is not supported by the printer the document is routed to.

Font Substitution by Font ID Range

Table 38 on page 416 divides font IDs into ranges. The ranges represent fonts of the same weight and size. A default font is selected in each range for substitution when a font is not found.

For example, one range in the table is all font IDs greater than 0 but less than or equal to 65. These fonts are all Roman medium, 10 pitch fonts. The default font is Courier Roman medium, 10 pitch.

By using this table, you can identify which font ID is substituted for any font ID that is not in Table 37 on page 404.

You need to use this table only if you cannot find your font ID in Table 37 on page 404.

Host Resident to Printer Resident Font Character Set Mapping

Table 39 on page 417 contains the substitutions made when your application specifies a font character set and you want to print the spooled file on an AFP-configured 4224, 4230, 4234, 4247, or 64xx printer.

These printers do not support downloading of font character sets. These printers use font IDs. When the printer writer program sends the spooled file to one of these printers, a substitution from font character sets to font global identifiers (FGIDs) takes place.

A substitution occurs if: the host resident font character set, or code page, cannot be found on the iSeries server, and your printer supports printer resident fonts as well as host resident fonts. The system substitutes from host resident font character set to FGID. This is true for the printers that are listed above except for the: 3112, 3116, 3130, 3160–001, 3812, 3816, 3912, 3916, 3930, 3935, 4028, 4312, 4317, and the 4324 printer. It is also true for the InfoPrint 20, InfoPrint 32, InfoPrint 60, InfoPrint 3000, and InfoPrint 4000 printers. This is true of host resident code pages to CPGIDs as well.

Go to Table 39 on page 417 to find out which FGID is substituted for each font character set.

Printer Resident to Host Resident Font Character Set Mapping

Table 40 on page 442 contains the substitutions made when your application specifies an FGID or font ID and you want to print the spooled file on a 3820, 3825, 3827, 3829, 3831, 3835, or 3900 Model 1 printer. These printers support only host font the case of the 38820 character sets.

The font character sets reside on the iSeries server. When the printer writer program sends the spooled file to one of these printers, a substitution from FGIDs to font character sets takes place.

Go to Table 40 on page 442 to find out which host font character set is substituted for each FGID.

Printer Resident to Host Resident Code Page Mapping

Table 41 on page 459 contains the substitutions made when your application specifies an OS/400 code page global identifier (CPGID) and you want to print the spooled file on a 3820, 3825, 3827, 3829, 3831, 3835 or 3900 printer.

These printers support code page names, but not CPGIDs.

When the printer writer program sends the spooled file to one of these printers, a substitution from CPGID to code page name takes place.

Go to Table 41 on page 459 to find out which code page name is substituted for each CPGID.

Character Identifier (CHRID) Values Supported

Table 42 on page 463 contains the character identifier (CHRID) values and supported printers for the different language groups.

Host Resident to Printer Resident Code Page Mapping

Table 43 on page 474 contains the substitutions made when your application specifies a code page name and you want to print the spooled file on an AFP-configured 4224, 4230, 4234, 4247, 6408, or 6412 printer.

These printers support CPGIDs, but not code page names.

When the printer writer program sends the spooled file to one of these printers, a substitution from code page name to CPGID takes place.

A substitution occurs if: the host resident font character set, or code page, cannot be found on the iSeries server and your printer supports printer resident fonts as well as host resident fonts. The system substitutes from host resident font character set to FGID. This is true for the printers that are listed above except for the: 3112, 3116, 3130, 3160–001, 3812, 3816, 3912, 3916, 3930, 3935, 4028, 4312, 4317, and the 4324 printer. It is also true for the InfoPrint 20, InfoPrint 32, InfoPrint 60, InfoPrint 3000, and InfoPrint 4000 printers. This is true of host resident code pages to CPGIDs as well.

Go to Table 39 on page 417 to find out which FGID is substituted for each font character set.

Go to Table 43 on page 474 to find out which CPGID is substituted for each code page name.

Lines Per Inch (LPI) Values Supported

Table 44 on page 476 contains the lines per inch (LPI) values supported for certain printers.

Characters Per Inch (CPI) Values Supported

Table 45 on page 477 contains the characters-per-inch (CPI) values supported for certain printers.

4019 Printer Information

Table 46 on page 478 contains specific information about the 4019 printer.

4234 Compressed Font Substitution

Table 47 on page 481 contains information about font substitution on 4234 printers when printing with a lines per inch (LPI) value greater than or equal to 8.

Font Attributes
Font attributes are the characteristics or properties that combine to give a font identity. For example: attributes can be 14 point (height of the font), bold, and italic.

Types of Fonts: The following diagram identifies the types of fonts and gives examples of each type:
- Mixed pitch fonts which simulate proportionally spaced fonts.

Characters in the font have a limited number of widths. Overall spacing is about 12 characters per inch. Examples are Document or Essay fonts.

- Uniformly spaced fonts which are similar to typewriter fonts.

  Characters in the font are all the same width. Examples are Courier and Gothic Text fonts. Some uniformly spaced fonts and many typographic fonts are scalable. For scalable fonts, specify a point size to indicate the size of the font. For example a 12 point uniformly spaced font corresponds to 10 CPI. An example of such a font is font 416, Courier Roman Medium. If no point size is specified when using scalable fonts, 10 point is defaulted.

- Typographic fonts

  Typographic fonts have variable height, measured in points (1 point = 1/72 inch). Therefore, a 36-point font has characters that are 1/2 inch high. Typographic fonts have variable widths. Width is part of the design and varies on a character-by-character basis. Examples are Sonoran Serif and Century Schoolbook.

Mixed Pitch

Printing on the AS/400

Uniformly Spaced
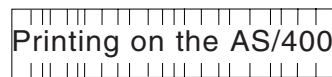
Printing on the AS/400

Typographic

Printing on the AS/400	6 pt Century Schoolbook
Printing on the AS/400	8 pt Century Schoolbook
Printing on the AS/400	10 pt Century Schoolbook

RV2H301-2

The legend and table below provide information about each font. This information could save you time in trial-and-error testing when choosing a font for your application.

Table Legend	
FGID	Font Global Identifier
Name	Name of Font
Font Type	**U** = Uniformly Spaced **M** = Mixed Pitch **T** = Typographic
Attributes	**Blank** = Roman **b** = Bold **i** = Italics **s** = Second Strike **w** = Double Wide
Point	Point size (Blank for uniformly spaced & mixed pitch fonts) S = Scalable

	Table Legend	
Pitch	Characters per inch (For example, 17)	
	S = Scalable	

Table 36. Font Information

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
2	Delegate	U			10
3	OCR-B	U			10
5	Rhetoric/Orator	U			
8	Scribe/Symbol	U			10
10	Cyrillic 22	U			10
11	Courier	U			10
12	Prestige	U			10
13	Artisan	U			10
18	Courier Italic	U	I		10
19	OCR-A	U			10
20	Pica	U			10
21	Katakana	U			10
25	Presentor	U			10
26	Matrix Gothic	U			10
30	Symbol	U			10
31	Aviv	U			10
36	Letter Gothic	U			10
38	Orator Bold	U	b		10
39	Gothic Bold	U	b		10
40	Gothic	U			10
41	Roman Text	U			10
42	Serif	U			10
43	Serif Italic	U	i		10
44	Katakana Gothic	U			10
46	Courier Bold	U	b		10
49	Shalom	U			10
50	Shalom Bold	U	b		10
51	Matrix Gothic	U			10
52	Courier	U			10
55	Aviv Bold	U	b		10
61	Nasseem	U			10
62	Nasseem Italic	U	i		10
63	Nasseem Bold	U	b		10
64	Nasseem Italic Bold	U	bi		10

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
66	Gothic	U		12	
68	Gothic Italic	U	i		12
69	Gothic Bold	U	b		12
70	Serif	U			12
71	Serif Italic	U	i		12
72	Serif Bold	U	b		12
74	Matrix Gothic	U			12
75	Courier	U			12
76	APL	U			12
78	Katakana	U			12
80	Symbol	U			12
84	Script	U			12
85	Courier	U			12
86	Prestige	U			12
87	Letter Gothic	U			12
91	Light Italic	U	i		12
92	Courier Italic	U	i		12
95	Adjudant	U			12
96	Old World	U			12
98	Shalom	U			12
99	Aviv	U			12
101	Shalom Bold	U	b		12
102	Aviv Bold	U	b		12
103	Nasseem	U			12
109	Letter Gothic Italic	U	i		12
110	Letter Gothic Bold	U	b		12
111	Prestige Bold	U	b		12
112	Prestige Italic	U	i		12
154	Essay	M			12
155	Boldface Italic	M	bi		12
157	Title	M			12
158	Modern	M			12
159	Boldface	M	b		12
160	Essay	M			12
162	Essay Italic	M	i		12
163	Essay Bold	M	b		12
164	Prestige	M			12
167	Barak	M			12
168	Barak Bold	M	b		12

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
173	Essay	M			12
174	Gothic	M			12
175	Document	M			12
178	Barak	M			18
179	Barak Bold	M	b		18
180	Barak	M			15
181	Barak Mixed Bold	M	b		15
182	Barak	M			5
183	Barak Bold	M	b		5
186	Press Roman	M			12
187	Press Roman Bold	M	b		12
188	Press Roman Italic	M	i		12
189	Press Roman Italic Bold	M	bi		12
190	Foundry	M			12
191	Foundry Bold	M	b		12
194	Foundry Italic	M	i		12
195	Foundry Italic Bold	M	bi		12
203	Data 1	U			13
204	Matrix Gothic	U			13
205	Matrix Gothic	U			13
211	Shalom	U			15
212	Shalom Bold	U	b		15
221	Prestige	U			15
222	Gothic	U			15
223	Courier	U			15
225	Symbol	U			15
226	Shalom	U			15
229	Serif	U			15
230	Gothic	U			15
232	Matrix Gothic	U			15
233	Matrix Courier	U			15
234	Shalom Bold	U	b		15
244	Courier Double Wide	U	w		5
245	Courier Bold Double Wide	U	wb		5
247	Shalom Bold	U	b		17
248	Shalom	U			17
249	Katakana	U			17
252	Courier	U			17

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
253	Courier Bold	U	b		17
254	Courier	U			17
255	Matrix Gothic	U			17
256	Prestige	U			17
258	Matrix Gothic	U			18
259	Matrix Gothic	U			18
279	Nasseem	U			17
281	Gothic Text	U			20
282	Aviv	U			20
283	Letter Gothic	U			20
285	Letter Gothic	U			25
290	Gothic Text	U			27
300	Gothic	U		S	17, S
304	Gothic Text	U		S	S
305	OCR-A	U		S	S
306	OCR-B	U		S	S
307	APL	U		S	S
318	Prestige Bold	U	b	S	S
319	Prestige Italic	U	i	S	S
322	APL Bold	U	b	S	S
400	Gothic	U		S	17, S
404	Letter Gothic Bold	U	b	S	S
416	Courier Roman Medium	U		S	S
420	Courier Roman Bold	U	b	S	S
424	Courier Roman Italic	U	i	S	S
428	Courier Roman Italic Bold	U	bi	S	S
432	Prestige	U		S	S
434	Orator Bold	U	b		8
435	Orator Bold	U	b		6
751	Sonoran Serif	T		8P.	27
752	Nasseem	T		12P	18
753	Nasseem Bold	T	b	12P	18
754	Nasseem Bold	T	b	18P	12
755	Nasseem Bold	T	b	24P	9
756	Nasseem Italic	T	i	12P	18
757	Nasseem Bold Italic	T	bi	12P	18
758	Nasseem Bold Italic	T	bi	18P	12
759	Nasseem Bold Italic	T	bi	24P	9

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
760	Times Roman	T		6P	36
761	Times Roman Bold	T	b	12P	18
762	Times Roman Bold	T	b	10P	15
763	Times Roman Italic	T	i	12P	18
764	Times Roman Bold Italic	T	bi	10P	21
765	Times Roman Bold Italic	T	bi	12P	18
1051	Sonoran Serif	T		10P	21
1053	Sonoran Serif Bold	T	b	10P	21
1056	Sonoran Serif Italic	T	i	10P	21
1351	Sonoran Serif	T		12P	18
1653	Sonoran Serif Bold	T	b		13
1803	Sonoran Serif Bold	T	b	18P	12
2103	Sonoran Serif Bold	T	b	24P	9
2304	Helvetica Roman Medium	T		S	S
2305	Helvetica Roman Bold	T	b	S	S
2306	Helvetica Roman Italic	T	i	S	S
2307	Helvetica Roman Italic Bold	T	bi	S	S
2308	Times New Roman Medium	T		S	S
2309	Times New Roman Bold	T	b	S	S
2310	Times New Roman Italic	T	i	S	S
2311	Times New Roman Italic Bold	T	bi	S	S
4407	Sonoran Serif	T		8P	*27
4407	Sonoran Serif	T		10P	*21
4407	Sonoran Serif	T		12P	*18
4427	Sonoran Serif Bold	T	b	10P	*21
4427	Sonoran Serif Bold	T	b	16P	*13
4427	Sonoran Serif Bold	T	b	24P	*9
4535	Sonoran Serif Italic	T	i	10P	*21
4919	Goudy	T		6P	*36
4919	Goudy	T		8P	*27
4919	Goudy	T		10P	*21
4919	Goudy	T		12P	*18
4939	Goudy Bold	T	T	10P	*21
4939	Goudy Bold	T	b	14P	*15

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
4939	Goudy Bold	T	b	18P	*12
5047	Goudy Italic	T	i	10P	*21
5067	Goudy Bold Italic	T	bi	10P	*21
5687	Times Roman	T		6P	*36
5687	Times Roman	T		8P	*27
5687	Times Roman	T		10P	*21
5687	Times Roman	T		12P	*18
5707	Times Roman Bold	T	b	10P	*21
5707	Times Roman Bold	T	b	12P	*18
5707	Times Roman Bold	T	b	14P	*15
5707	Times Roman Bold	T	b	18P	*12
5707	Times Roman Bold	T	b	24P	*12
5815	Times Roman Italic	T	i	10P	*21
5815	Times Roman Italic	T	i	12P	*18
5835	Times Roman Italic Bold	T	bi	10P	*21
5835	Times Roman Italic Bold	T	bi	12P	*18
5943	University	T		12P	*18
5943	University	T		14P	*15
5943	University	T		18P	*12
6199	Palatino	T		6P	*36
6199	Palatino	T		8P	*27
6199	Palatino	T		10P	*21
6199	Palatino	T		12P	*18
6219	Palatino Bold	T	b	10P	*21
6219	Palatino Bold	T	b	14P	*15
6219	Palatino Bold	T	b	18P	*12
6327	Palatino Italic	T	i	10P	*21
6347	Palatino Italic Bold	T	bi	10P	*21
8503	Baskerville	T		6P	*36
8503	Baskerville	T		8P	*27
8503	Baskerville	T		10P	*21
8503	Baskerville	T		12P	*18
8523	Baskerville Bold	T	b	10P	*21
8523	Baskerville Bold	T	b	14P	*15
8523	Baskerville Bold	T	b	18P	*12
8631	Baskerville Italic	T	i	10P	*21
8651	Baskerville Italic Bold	T	bi	10P	*21
8759	Nasseem	T		12P	*18

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
8779	Nasseem Bold	T	b	12P	*18
8779	Nasseem Bold	T	b	18P	*12
8779	Nasseem Bold	T	b	24P	*9
8887	Nasseem Italic	T	i	12P	*18
8907	Nasseem Italic Bold	T	bi	12P	*18
8907	Nasseem Italic Bold	T	bi	18P	*12
8907	Nasseem Italic Bold	T	bi	24P	*9
12855	Narkisim	T		8P	*27
12855	Narkisim	T		10P	*21
12855	Narkisim	T		18P	*12
12855	Narkisim	T		24P	*9
12875	Narkisim Bold	T	b	8P	*27
12875	Narkisim Bold	T	b	10P	*21
12875	Narkisim Bold	T	b	12P	*18
16951	Century Schoolbook	T		6P	*36
16951	Century Schoolbook	T		8P	*27
16951	Century Schoolbook	T		10P	*21
16951	Century Schoolbook	T		12P	*18
16971	Century Schoolbook Bold	T	b	10P	*21
16971	Century Schoolbook Bold	T	b	14P	*15
16971	Century Schoolbook Bold	T	b	18P	*12
17079	Century Schoolbook Italic	T	i	10P	*21
17099	Century Schoolbook Italic Bold	T	bi	10P	*21
20224	Boldface	T	b	S	S
33335	Optima	T		6P	*36
33335	Optima	T		8P	*27
33335	Optima	T		10P	*21
33335	Optima	T		12P	*18
33355	Optima Bold	T	b	10P	*21
33355	Optima Bold	T	b	14P	*15
33355	Optima Bold	T	b	18P	*12
33463	Optima Italic	T	i	10P	*21
33483	Optima Italic Bold	T	bi	10P	*21
33591	Futura	T		6P	*36
33591	Futura	T		8P	*27
33591	Futura	T		10P	*21

Table 36. Font Information (continued)

FGID	Name	Type of Font	Attributes	Point	Pitch (CPI)
33591	Futura	T		12P	*18
33601	Futura Bold	T	b	10P	*21
33601	Futura Bold	T	b	14P	*15
33601	Futura Bold	T	b	18P	*12
33719	Futura Italic	T	i	10P	*21
33729	Futura Italic Bold	T	bi	10P	*21
34103	Helvetica	T		6P	*36
34103	Helvetica	T		8P	*27
34103	Helvetica	T		10P	*21
34103	Helvetica	T		12P	*18
34123	Helvetica Bold	T	b	10P	*21
34123	Helvetica Bold	T	b	14P	*15
34123	Helvetica Bold	T	b	18P	*12
34231	Helvetica Italic	T	i	10P	*21
34251	Helvetica Italic Bold	T	bi	10P	*21
37431	Old English	T		12P	*18
37431	Old English	T		14P	*15
37431	Old English	T		18P	*12
41783	Coronet Cursive	T		12P	*18
41803	Coronet Cursive Bold	T	b	14P	*15
41803	Coronet Cursive Bold	T	b	18P	*12

Note: Pitch or CPI column for typographic fonts indicates the width of the space character between printed characters. Width, pitch, and CPI of other space characters will vary.

Font Substitution

Font substitution is done by the iSeries server when the application specifies a font ID that is not supported by the designated printer or cannot be downloaded from the system to the designated printer.

Table 37 on page 404 lists many fonts (by FGID number) and printers that are supported. A blank in any column indicates that the font ID is supported by that printer, and no substitution takes place. However, if your application specifies a font ID that is not in the table, you need to refer to Table 38 on page 416. Table 38 on page 416 provides the substituted FGID for font IDs in ranges such as FGID 0 through FGID 65.

How To Use the Font Substitution Charts: Following are three examples to familiarize you with font substitution on the iSeries server.

- Example one shows how to verify whether or not your font ID is supported by a certain printer.
- Example two shows how to find out what font ID the system substitutes if the printer you want to use does not support your font ID.
- Example three shows how to find out what font ID the system substitutes if your font ID is not available on the system or on the printer.

Example One: If you want to verify that a font ID is supported by a certain printer, locate the font ID in Table 37 on page 404. For example, locate font ID 112. Font ID 112 is supported by the 3812 and 3816 SCS and IPDS printers and the 4028 printer (this is indicated by blanks in those spaces). The 4019 printer supports font ID 112 on a font card resident in the 4019 printer. The 4224, 4234, and 5219 printers substitute font ID 87 or 86.

Note: A font card is a hardware card that can have many font character sets resident on it. Font cards can be installed in printers to provide additional fonts.

Example Two: If your application uses a font ID that is not supported on all printers, you can determine the substitution by looking in Table 37 on page 404. For example, locate font ID 30. The table shows that font ID 30 is supported on the 3812 and 3816 SCS and IPDS printers. However, if you are using any of the other printers listed in the table, font ID 11 is substituted for font ID 30.

Example Three: Let us say your application calls for font ID 4 and you want to print the spooled file on a 4224 printer. To determine if font ID 4 is a supported font or one that is substituted for, read through the following steps:

Step 1   Look in Table 37 on page 404 to see if font ID 4 is listed. Font ID 4 is not in Table 37 on page 404.

Step 2   Next, look in Table 38 on page 416. The table shows that font ID 11 is substituted for fonts 0 through 65.

Step 3   Return to Table 37 on page 404 and locate font ID 11. This table shows that font ID 11 is supported on the 4224 printer.

Step 4   The result of the font ID substitution is that your application will print using font ID 11.

Changing Font IDs: To permanently change the font ID, you could, in your application, specify a different font ID or use the Change Printer File (CHGPRTF) command to specify a new font ID for the printer file. Information in Table 36 on page 395 ID.

To temporarily change the font ID for your application, you could override the font selection in your printer file by using the Override with Printer File (OVRPRTF) command before the application runs.

Font Substitution and the 4019 Printer: The 4019 printer is supported by the iSeries server, as an emulated printer (usually 3812 or 5219). The system treats the device as a physical 3812 or 5219. Therefore, the font support and font substitution of the emulated printer is used. This emulation limits access to some of the 4019 fonts.

To access most of the 4019-supported fonts, an IBM-supplied program named QWP4019 is available. QWP4019 sets a flag in the emulated printer's device description to inform the system to use the 4019 font tables.

For more information and examples on how the QWP4019 program works, see Table 30 on page 312.

Note to Reader:

An asterisk is used in the following chart to indicate that the substituted font may have a different pitch. When the substituted font is scalable, the pitch is the same.

Table 37. Font Substitution

	Printers							
FGID	**4224 4230 IPDS**	**4234 IPDS**	**3812 or 3816 SCS**	**3812 or 3816 IPDS**	**5219**	**3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32**	**4019[1]**	**3130 3160 3935 InfoPrint 3000 InfoPrint 4000**
2	11	11	11	11	11	11[2]		11
3					11			
5	11	26				11[2]		11
8	11	11	11	11	11	11		11
10	11	11	11	11	11	11		*416
11								
12	11	26						
13	11	11				11	11	11
18	11	26			11			
19					11			
20	11	26				11	11	*432
21	11	11	11	11	11	11		*304
25	11	11	11	11	11	11[2]		11
26						11	11	11
30	11	11			11	11	11	11
31	26	26	26		26	11	11	*416
36	11	11	11	11	11	11[2]		*400
38	11	26			11	46	46	46
39	26	26			11	46	46	46
40	26	26			11	11	11	*304
41	11	26			11	11	11	11
42	11	26			11	11	11	11
43	11	26			11	18	11	18
44	11	11			11	11	11	*304
46	11	26			11			
49	26	26	26		26	11		*416
50	26	26			26	46		*420
51	26				26	11	11	11

Table 37. Font Substitution (continued)

FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
						Printers		
52	11				11	11	11	11
55	26	26	26		26	46	46	*420
61	11	11	11	11	11	11		*416
62	11	11	11	11	11	18		*424
63	11	11	11	11	11	46		*420
64	11	11	11	11	11	46		*428
66	87	87			87	85	85	*304
68	87	87			87	92	85	92
69	87	87			87	111	85	111
70	87	87			87	85	85	85
71	87	87			87	92	85	92
72	87	87			87	111	85	111
74	87		87	87	87	85	85	85
75	85		85	85	85	85	85	85
76	85	85	85	85	85			
78	85	85	85	85	85	85		*304
80	87	87				85		85
84	87	87				85[2]		85
85								
86	87	87						
87						85[2]		*400
91	87	87				92[2]		92
92	85	85	85	85	85			
95	85	85	85	85	85	85[2]		85
96	85	85	85	85	85	85[2]		85
98	87	87	87		87	85		*416
99	87	87	87		87	85	85	*416
101	87	87	87		87	111	85	*416
102	87	87	87		87	111	85	*420
103	85	85	85	85	85	85		*416
109	85	85	85	85	85	92[2]		92

Table 37. Font Substitution (continued)

FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
					Printers			
110	87	87			87	11[2]		*404
111	87	87			86			
112	87	87			86			
154	85		160	160	160	164	159	159
155	160	160			160	159[2]		159
157	160	160	160	160	160	164[2]		159
158	160	160				164[2]		159
159	160	160						
160						164[2]		159
162	160	160				164[2]		159
163	160	160			160	159	159	159
164	160	160	160	160	160			159
167	160	160	160		160	164		*416
168	160	160	160		160	159	159	*420
173	160	160			160	164	159	159
174	160	160	160	160	160	164	159	159
175	160	160			160	164	159	159
178	*400	*258	*281		*222	*281	*254	*416
179	*400	*258	*281		*222	*281	*254	*420
180	*222	*222	*230		*222	*223	*254	*416
181	*222	*222	*230		*222	*223	*254	*420
182	*11	*11	*244		*11	*11	*11	*416
183	*11	*11	*244		*11	*46	*46	*420
186	160	160	160	160	160	164[2]		159
187	160	160	160	160	160	159[2]		159
188	160	160	160	160	160	164[2]		159
189	160	160	160	160	160	159[2]		159
190	160	160	160	160	160	164[2]		159
191	160	160	160	160	160	159[2]		159
194	160	160	160	160	160	164[2]		159
195	160	160	160	160	160	159[2]		159

Table 37. Font Substitution (continued)

FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
						Printers		
203	*222	204	204	204	*222		*254	*416
204	*222				*222	*223	*254	*304
205	*222		204	204	*222	*223	*254	*416
211	222	222	230		222	223	*254	*416
212	222	222	230		222	223	*254	*420
221	222	222						
222			230	230		223[2]		*304
223								
225	222	222				223	*254	223
226	222	222	230		222	223		*416
229	222	222			222	223	*254	223
230	222	222			222	223	*254	*304
232	222		230	230	222	223	*254	*223
233	223		230	230	223	223	*254	223
234	222	222	230		222	223	*254	*420
244	*11	*26			*11	*11		*416
245	*11	*26			*11	*46		*420
247	*400	*258	252		*222	254	254	*420
248	*400	*258	252		*222	254	254	*416
249	*400	*258	252	252	*222	254		*304
252	*400	*258			*222	254	254	254
253	*400	*258			*222	254	254	*420
254	*400	*258			*222.			
255	*400	*258	252	252	*222	254		254
256	*400	*258	252	252	*222.			
258	*400		*281	*281	*222	*281	*254	*416
259	*400		*281	*281	*222	*281	*254	*416
279	*400	*258	252	252	*222	254		*416
281	*400	*258			*222			
282	*400	*258	281		*222	281		*416
283	*400	*258	281	281.	*222	281		*400

Table 37. Font Substitution (continued)

					Printers			
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
285	*400	*258	*290	*290	*222	281[2]		*400
290	*400	*258			*222		*254	*416
300	400		*252	*252	*222	*254	*254	*416
304	*26	*26	*26	*26	*26	*11	*11	
305	*19	*19	*19	*19	*11	*19	*19	
306	*3	*3	*3	*3	*11	*3	*3	
307	*85	*85	*85	*85	*85	*76	*76	
318	*11	*26	*12	*12	*12	*12	*12	
319	*11	*26	*12	*12	*12	*12	*12	
322	*85	*85	*85	*85	*85	*76	*76	
400			*252	*252	*222	*254	*254	
404	*26	*26	*39	*39	*11	*46	*46	
416	*11	*11	*11	*11	*11	*11	*11	
420	*11	*26	*46	*46	*11	*46	*46	
424	*11	*26	*18	*18	*11	*18	*18	
428	*11	*26	*18	*18	*11	*18	*18	
432	*11	*26	*12	*12	*12	*12	*12	
434	*11	*11	*11	*11	*11	46[2]		*420
435	*11	*11	*11	*11	*11	46[2]		*420
751	*400	*258			*222.		*254	
752	*400	*258	*281	*281	*222		*254	*2308
753	*400	*258	*281	*281	*222		*254	*2309
754	*85	*85	*85	*85	*85		*85	*2309
755	*11	*11	*11	*11	*11		*46	*2309
756	*400	*258	*281	*281	*222		*254	*2310
757	*400	*258	*281	*281	*222		*254	*2311
758	*85	*85	*85	*85	*85		*85	*2311
759	*11	*11	*11	*11	*11		*46	*2311
760	*400	*258	*290	*290	*222		*254	
761	*400	*258	*281	*281	*222		*254	
762	*222	*222	*230	*230	*222		*254	

Table 37. Font Substitution (continued)

FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
				Printers				
763	*400	*258	*281	*281	*222		*254	
764	*400	*258	*290	*290	*222		*254	
765	*400	*258	*281	*281	*222		*254	
1051	*400	*258			*222		*254	
1053	*400	*258			*222		*254	
1056	*400	*258			*222		*254	
1351	*400	*258			*222		*254	
1653	*222	*222			*222		*254	*2309
1803	*85	*85	*85	*85	*85		*85	
2103	*11	*11			*11		*46	
2304	*400	*258	*290	*290	*222	*760	*254	
2305	*400	*258	*281	*281	*222	*761	*254	
2306	*400	*258	*281	*281	*222	*763	*254	
2307	*400	*258	*280	*290	*222	*764	*254	
2308	*400	*258	*290	*290	*222	*760	*254	
2309	*400	*258	*281	*281	*222	*761	*254	
2310	*400	*258	*281	*281	*222	*763	*254	
2311	*400	*258	*290	*290	*222	*764	*254	
4407 (8P)	*400	*258			*222	5687[2]	*254	*2308
4407 (10P)	*400	*258			*222	5687[2]	*254	*2308
4407 (12P)	*400	*258			*222	5687[2]	*254	*2308
4427 (10P)	*400	258			*222	5687[2]	*254	*2309
4427 (16P)	*222	*222			*11	*5707	*254	*2309
4427 (24P)	*11	*11			*11	5707[2]	*254	*2309
4535 (10P)	*400	*258			*222	5687[2]	*46	*2310
4919 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308

Table 37. Font Substitution (continued)

	Printers							
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
4919 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
4919 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
4919 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
4939 (10P)	*400	*258	*1053	*1053	*222	5707[2]		*2309
4939 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
4939 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
5047 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2310
5067 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2311
5687 (6P)	*400	*258	*290	*290	*222			
5687 (8P)	*400	*258	*751	*751	*222			
5687 (10P)	*400	*258	*1051	*1051	*222			
5687 (12P)	*400	*258	*1351	*1351	*222			
5707 (10P)	*400	*258	*1053	*1053	*222			
5707 (12P)	*400	*258	*1351	*1351	*222		*254	
5707 (14P)	*222	*222	*1351	*1351	*222			
5707 (18P)	*85	*85	*1653	*1653	*85			
5707 (24P)	*11	*11	*2103	*2103	*11			
5815 (10P)	*400	*258	*1056	*1056	*222			

Table 37. Font Substitution (continued)

FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
5815 (12P)	*400	*258	*1351	*1351	*222		*254	
5835 (10P)	*400	*258	*1053	*1053	*222			
5835 (12P)	*400	*258	*1351	*1351	*222			
5943 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
5943 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2308
5943 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2308
6199 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308
6199 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
6199 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
6199 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
6219 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
6219 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
6219 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
6327 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2310
6347 (10P)	*400	*258	*1053	*1053	*222	5686[2]		*2311
8503 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308
8503 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
8503 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308

Table 37. Font Substitution (continued)

	Printers							
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
8503 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
8523 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
8523 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
8523 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
8631 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2310
8651 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2311
8759 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
8779 (12P)	*400	*258	*1351	*1351	*222	5707[2]		*2309
8779 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
8779 (24P)	*11	*11	*2103	*2103	*11	5707[2]		*2309
8887 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2310
8907 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2311
8907 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2311
8907 (24P)	*11	*11	*2103	*2103	*11	5707[2]		*2311
12855 (8P)	*400	*258	*751		*222	5687[2]		*2308
12855 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
12855 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2308
12855 (24P)	*11	*11	*2103	*2103	*11	5707[2]		*2308

Table 37. Font Substitution (continued)

	Printers							
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
12875 (8P)	*400	*258	*751		*222	5687[2]		*2309
12875 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
12875 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2309
16951 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308
16951 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
16951 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
16951 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
16971 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
16971 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
16971 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
17079 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2311
17099 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2311
20224	*160	*160	*159	*159	*159	*159	*159	
33335 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308
33335 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
33335 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
33335 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
33355 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
33355 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309

Table 37. Font Substitution (continued)

	Printers							
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
33355 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
33463 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2310
33483 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2311
33591 (6P)	*400	*258	*290	*290	*222	5687[2]		*2308
33591 (8P)	*400	*258	*751	*751	*222	5687[2]		*2308
33591 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2308
33591 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
33601 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2309
33601 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
33601 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309
33719 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2310
33729 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2311
34103 (6P)	*400	*258	*290	*290	*222	5687[2]		*2304
34103 (8P)	*400	*258	*751	*751	*222	5687[2]		*2304
34103 (10P)	*400	*258	*1051	*1051	*222	5687[2]		*2304
34103 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2304
34123 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2305
34123 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2305

Table 37. Font Substitution (continued)

	Printers							
FGID	4224 4230 IPDS	4234 IPDS	3812 or 3816 SCS	3812 or 3816 IPDS	5219	3112 3116 3912 3916 4028 4312 4317 4324 InfoPrint 20, InfoPrint 32	4019[1]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
34123 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2305
34231 (10P)	*400	*258	*1056	*1056	*222	5687[2]		*2306
34251 (10P)	*400	*258	*1053	*1053	*222	5687[2]		*2307
37431 (12)	*400	*258	*1351	*1351	*222	5687[2]		*2308
37431 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2308
37431 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2308
41783 (12P)	*400	*258	*1351	*1351	*222	5687[2]		*2308
41803 (14P)	*222	*222	*1351	*1351	*222	5707[2]		*2309
41803 (18P)	*85	*85	*1653	*1653	*85	5707[2]		*2309

:

[1]   The 4019 printer has five resident fonts: FGID 11, 46, 85, 159, and 254. The iSeries server sends any of those FGIDs that do not show a substitution in Table 37 on page 404 to the emulator that the 4019 is attached to. The emulator may not support all of the FGIDs and may report an error or perform a substitution of its own.

[2]   The 4028 performs the font substitution as shown unless a font card has been installed that contains that FGID. For example, if a font card with an FGID of 2 is installed, the iSeries server sends the FGID of 2 to the printer. However, if the font card is not installed, the system substitutes an FGID of 11.

Font Substitution by Font ID Range

If your application specifies a font ID that is not found in Table 37 on page 404 or is not resident in the printer (font card), the system makes a substitution based on the font ID ranges in the following table. For example, if font ID 4 is specified in your application, the iSeries server substitutes font ID 11 as shown in the table below.

Table 38. Font Substitution by Font ID Range.

FGID		Substituted FGID
Fonts 0 through 65		11
Fonts 66 through 153		85
Fonts 154 through 200		160
Fonts 201 through 210		204
Fonts 211 through 239		223
Fonts 240 through 246		245
Fonts 247 through 257		252
Fonts 258 through 259		259
Fonts 260 through 273		434
Fonts 274 through 279		279
Fonts 280 through 284		281
Fonts 285 through 289		285
Fonts 290 through 299		290
Fonts 300 through 511		252
Fonts 512 through 2303		252
Fonts 2304 through 3839 or Fonts 4069 through 65279	Fonts with point size equal to 0 or not specified	252
	Fonts with point size greater than 0 but less than 7.6	5687-6p
	Fonts with point size greater than or equal to 7.6 but less than 9.6	5687-8p
	Fonts with point size greater than or equal to 9.6 but less than 11.6	5687-10p
	Fonts with point size greater than or equal to 11.6 but less than 13.6	5687-12p
	Fonts with point size greater than or equal 13.6 but less than 17.6	5707-14p
	Fonts with point size greater than or equal to 17.6 but less than 23.6	5707-18p
	Fonts with point size greater than or equal to 23.6	5707-24p
Fonts 3840 through 4095 (User-defined)		No Substitution
Fonts 65280 through 65534 (User-definded)		No Substitution

Host Resident to Printer Resident Font Character Set Mapping

If your application specifies a host resident font character set (fonts are stored on the iSeries server) and you want to print the spooled file on an AFP-configured 4224, 4230, 4234, or 64xx printer, the system must substitute a printer resident font character set (fonts stored on the printer).

The following table can help you determine what printer resident font character set is requested when your spooled file references a host resident font character set instead of a registered font identifier (ID). This font substitution is necessary because these printers do not support the downloading of 240-pel host resident font character sets. Depending upon the host resident font character set that is

requested, the appropriate registered ID value, font width value, and font attributes are selected to match (as closely as possible) your font request.

The font width specifies the width of the blank character in 1440ths of an inch. This is an indicator of how many characters will fit per inch of space on the paper.

The Map Fidelity indicates whether or not the substituted printer resident font is considered to be an exact match to the font character set that is requested in your spooled file.

Table 39. Host Resident to Printer Resident Font Character Set Mapping

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0A053A0	33077	73	Normal	Exact
C0A053B0	33077	80	Normal	Exact
C0A053D0	33077	93	Normal	Exact
C0A053F0	33077	107	Normal	Exact
C0A053H0	33077	120	Normal	Exact
C0A053J0	33077	133	Normal	Exact
C0A053N0	33077	160	Normal	Exact
C0A053T0	33077	200	Normal	Exact
C0A053Z0	33077	240	Normal	Exact
C0A05300	33077	67	Normal	Exact
C0A05360	33077	40	Normal	Exact
C0A05370	33077	47	Normal	Exact
C0A05380	33077	53	Normal	Exact
C0A05390	33077	60	Normal	Exact
C0A055A0	33079	73	Normal	Exact
C0A055B0	33079	80	Normal	Exact
C0A055B1	33079	320	Normal	Exact
C0A055D0	33079	93	Normal	Exact
C0A055F0	33079	107	Normal	Exact
C0A055H0	33079	120	Normal	Exact
C0A055J0	33079	133	Normal	Exact
C0A055N0	33079	160	Normal	Exact
C0A055N1	33079	400	Normal	Exact
C0A055T0	33079	200	Normal	Exact
C0A055Z0	33079	240	Normal	Exact
C0A055Z1	33079	480	Normal	Exact
C0A05500	33079	67	Normal	Exact
C0A05560	33079	40	Normal	Exact
C0A05570	33079	47	Normal	Exact
C0A05580	33079	53	Normal	Exact
C0A05590	33079	60	Normal	Exact
C0A057A0	33081	73	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0A057B0	33081	80	Normal	Exact
C0A057D0	33081	93	Normal	Exact
C0A057F0	33081	107	Normal	Exact
C0A057H0	33081	120	Normal	Exact
C0A057J0	33081	133	Normal	Exact
C0A057N0	33081	160	Normal	Exact
C0A057T0	33081	200	Normal	Exact
C0A057Z0	33081	240	Normal	Exact
C0A05700	33081	67	Normal	Exact
C0A05760	33081	40	Normal	Exact
C0A05770	33081	47	Normal	Exact
C0A05780	33081	53	Normal	Exact
C0A05790	33081	60	Normal	Exact
C0A073A0	33097	73	Bold	Exact
C0A073B0	33097	80	Bold	Exact
C0A073D0	33097	93	Bold	Exact
C0A073F0	33097	107	Bold	Exact
C0A073H0	33097	120	Bold	Exact
C0A073J0	33097	133	Bold	Exact
C0A073N0	33097	160	Bold	Exact
C0A073T0	33097	200	Bold	Exact
C0A073Z0	33097	240	Bold	Exact
C0A07300	33097	67	Bold	Exact
C0A07360	33097	40	Bold	Exact
C0A07370	33097	47	Bold	Exact
C0A07380	33097	53	Bold	Exact
C0A07390	33097	60	Bold	Exact
C0A075A0	33099	73	Bold	Exact
C0A075B0	33099	80	Bold	Exact
C0A075B1	33099	320	Bold	Exact
C0A075D0	33099	93	Bold	Exact
C0A075F0	33099	107	Bold	Exact
C0A075H0	33099	120	Bold	Exact
C0A075J0	33099	133	Bold	Exact
C0A075N0	33099	160	Bold	Exact
C0A075N1	33099	400	Bold	Exact
C0A075T0	33099	200	Bold	Exact
C0A075Z0	33099	240	Bold	Exact
C0A075Z1	33099	480	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0A07500	33099	67	Bold	Exact
C0A07560	33099	40	Bold	Exact
C0A07570	33099	47	Bold	Exact
C0A07580	33099	53	Bold	Exact
C0A07590	33099	60	Bold	Exact
C0A077A0	33101	73	Bold	Exact
C0A077B0	33101	80	Bold	Exact
C0A077D0	33101	93	Bold	Exact
C0A077F0	33101	107	Bold	Exact
C0A077H0	33101	120	Bold	Exact
C0A077J0	33101	133	Bold	Exact
C0A077N0	33101	160	Bold	Exact
C0A077T0	33101	200	Bold	Exact
C0A077Z0	33101	240	Bold	Exact
C0A07700	33101	67	Bold	Exact
C0A07760	33101	40	Bold	Exact
C0A07770	33101	47	Bold	Exact
C0A07780	33101	53	Bold	Exact
C0A07790	33101	60	Bold	Exact
C0A153A0	33205	73	Italic	Exact
C0A153B0	33205	80	Italic	Exact
C0A153D0	33205	93	Italic	Exact
C0A153F0	33205	107	Italic	Exact
C0A153H0	33205	120	Italic	Exact
C0A153J0	33205	133	Italic	Exact
C0A153N0	33205	160	Italic	Exact
C0A153T0	33205	200	Italic	Exact
C0A153Z0	33205	240	Italic	Exact
C0A15300	33205	67	Italic	Exact
C0A15360	33205	40	Italic	Exact
C0A15370	33205	47	Italic	Exact
C0A15380	33205	53	Italic	Exact
C0A15390	33205	60	Italic	Exact
C0A155A0	33207	73	Italic	Exact
C0A155B0	33207	80	Italic	Exact
C0A155B1	33207	320	Italic	Exact
C0A155D0	33207	93	Italic	Exact
C0A155F0	33207	107	Italic	Exact
C0A155H0	33207	120	Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0A155J0	33207	133	Italic	Exact
C0A155N0	33207	160	Italic	Exact
C0A155N1	33207	400	Italic	Exact
C0A155T0	33207	200	Italic	Exact
C0A155Z0	33207	240	Italic	Exact
C0A155Z1	33207	480	Italic	Exact
C0A15500	33207	67	Italic	Exact
C0A15560	33207	40	Italic	Exact
C0A15570	33207	47	Italic	Exact
C0A15580	33207	53	Italic	Exact
C0A15590	33207	60	Italic	Exact
C0A175A0	33227	73	Bold Italic	Exact
C0A175B0	33227	80	Bold Italic	Exact
C0A175B1	33227	320	Bold Italic	Exact
C0A175D0	33227	93	Bold Italic	Exact
C0A175F0	33227	107	Bold Italic	Exact
C0A175H0	33227	120	Bold Italic	Exact
C0A175J0	33227	133	Bold Italic	Exact
C0A175N0	33227	160	Bold Italic	Exact
C0A175N1	33227	400	Bold Italic	Exact
C0A175T0	33227	200	Bold Italic	Exact
C0A175Z0	33227	240	Bold Italic	Exact
C0A175Z1	33227	480	Bold Italic	Exact
C0A17500	33227	67	Bold Italic	Exact
C0A17560	33227	40	Bold Italic	Exact
C0A17570	33227	47	Bold Italic	Exact
C0A17580	33227	53	Bold Italic	Exact
C0A17590	33227	60	Bold Italic	Exact
C0BPOSA0	323	120	Normal	Exact
C0BPOSBN	323	240	Normal	Exact
C0BPOSB0	323	168	Normal	Exact
C0BPOS91	323	144	Normal	Exact
C0B20CA0	335	73	Normal	Exact
C0B20CB0	335	80	Normal	Exact
C0B20CD0	335	93	Normal	Exact
C0B20CF0	335	107	Normal	Exact
C0B20CH0	335	120	Normal	Exact
C0B20CJ0	335	133	Normal	Exact
C0B20CN0	335	160	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0B20CT0	335	200	Normal	Exact
C0B20CZ0	335	240	Normal	Exact
C0B20C00	335	67	Normal	Exact
C0B20C50	335	33	Normal	Exact
C0B20C60	335	40	Normal	Exact
C0B20C70	335	47	Normal	Exact
C0B20C80	335	53	Normal	Exact
C0B20C90	335	60	Normal	Exact
C0B200A0	335	73	Normal	Exact
C0B200B0	335	80	Normal	Exact
C0B200D0	335	93	Normal	Exact
C0B200F0	335	107	Normal	Exact
C0B200H0	335	120	Normal	Exact
C0B200J0	335	133	Normal	Exact
C0B200N0	335	160	Normal	Exact
C0B200T0	335	200	Normal	Exact
C0B200Z0	335	240	Normal	Exact
C0B20000	335	67	Normal	Exact
C0B20050	335	33	Normal	Exact
C0B20060	335	40	Normal	Exact
C0B20070	335	47	Normal	Exact
C0B20080	335	53	Normal	Exact
C0B20090	335	60	Normal	Exact
C0B30CA0	337	73	Italic	Exact
C0B30CB0	337	80	Italic	Exact
C0B30CD0	337	93	Italic	Exact
C0B30CF0	337	107	Italic	Exact
C0B30CH0	337	120	Italic	Exact
C0B30CJ0	337	133	Italic	Exact
C0B30CN0	337	160	Italic	Exact
C0B30CT0	337	200	Italic	Exact
C0B30CZ0	337	240	Italic	Exact
C0B30C00	337	67	Italic	Exact
C0B30C50	337	33	Italic	Exact
C0B30C60	337	40	Italic	Exact
C0B30C70	337	47	Italic	Exact
C0B30C80	337	53	Italic	Exact
C0B30C90	337	60	Italic	Exact
C0B300A0	337	73	Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0B300B0	337	80	Italic	Exact
C0B300D0	337	93	Italic	Exact
C0B300F0	337	107	Italic	Exact
C0B300H0	337	120	Italic	Exact
C0B300J0	337	133	Italic	Exact
C0B300N0	337	160	Italic	Exact
C0B300T0	337	200	Italic	Exact
C0B300Z0	337	240	Italic	Exact
C0B30000	337	67	Italic	Exact
C0B30050	337	33	Italic	Exact
C0B30060	337	40	Italic	Exact
C0B30070	337	47	Italic	Exact
C0B30080	337	53	Italic	Exact
C0B30090	337	60	Italic	Exact
C0B40CA0	336	73	Bold	Exact
C0B40CB0	336	80	Bold	Exact
C0B40CD0	336	93	Bold	Exact
C0B40CF0	336	107	Bold	Exact
C0B40CH0	336	120	Bold	Exact
C0B40CJ0	336	133	Bold	Exact
C0B40CN0	336	160	Bold	Exact
C0B40CT0	336	200	Bold	Exact
C0B40CZ0	336	240	Bold	Exact
C0B40C00	336	67	Bold	Exact
C0B40C50	336	33	Bold	Exact
C0B40C60	336	40	Bold	Exact
C0B40C70	336	47	Bold	Exact
C0B40C80	336	53	Bold	Exact
C0B40C90	336	60	Bold	Exact
C0B400A0	336	73	Bold	Exact
C0B400B0	336	80	Bold	Exact
C0B400D0	336	93	Bold	Exact
C0B400F0	336	107	Bold	Exact
C0B400H0	336	120	Bold	Exact
C0B400J0	336	133	Bold	Exact
C0B400N0	336	160	Bold	Exact
C0B400T0	336	200	Bold	Exact
C0B400Z0	336	240	Bold	Exact
C0B40000	336	67	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0B40050	336	33	Bold	Exact
C0B40060	336	40	Bold	Exact
C0B40070	336	47	Bold	Exact
C0B40080	336	53	Bold	Exact
C0B40090	336	60	Bold	Exact
C0B50CA0	338	73	Bold Italic	Exact
C0B50CB0	338	80	Bold Italic	Exact
C0B50CD0	338	93	Bold Italic	Exact
C0B50CF0	338	107	Bold Italic	Exact
C0B50CH0	338	120	Bold Italic	Exact
C0B50CJ0	338	133	Bold Italic	Exact
C0B50CN0	338	160	Bold Italic	Exact
C0B50CT0	338	200	Bold Italic	Exact
C0B50CZ0	338	240	Bold Italic	Exact
C0B50C00	338	67	Bold Italic	Exact
C0B50C50	338	33	Bold Italic	Exact
C0B50C60	338	40	Bold Italic	Exact
C0B50C70	338	47	Bold Italic	Exact
C0B50C80	338	53	Bold Italic	Exact
C0B50C90	338	60	Bold Italic	Exact
C0B500A0	338	73	Bold Italic	Exact
C0B500B0	338	80	Bold Italic	Exact
C0B500D0	338	93	Bold Italic	Exact
C0B500F0	338	107	Bold Italic	Exact
C0B500H0	338	120	Bold Italic	Exact
C0B500J0	338	133	Bold Italic	Exact
C0B500N0	338	160	Bold Italic	Exact
C0B500T0	338	200	Bold Italic	Exact
C0B500Z0	338	240	Bold Italic	Exact
C0B50000	338	67	Bold Italic	Exact
C0B50050	338	33	Bold Italic	Exact
C0B50060	338	40	Bold Italic	Exact
C0B50070	338	47	Bold Italic	Exact
C0B50080	338	53	Bold Italic	Exact
C0B50090	338	60	Bold Italic	Exact
C0B60CA0	339	73	Reverse	Exact
C0B60CB0	339	80	Reverse	Exact
C0B60CD0	339	93	Reverse	Exact
C0B60CF0	339	107	Reverse	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0B60CH0	339	120	Reverse	Exact
C0B60CJ0	339	133	Reverse	Exact
C0B60CN0	339	160	Reverse	Exact
C0B60CT0	339	200	Reverse	Exact
C0B60CZ0	339	240	Reverse	Exact
C0B60C00	339	67	Reverse	Exact
C0B60C50	339	33	Reverse	Exact
C0B60C60	339	40	Reverse	Exact
C0B60C70	339	47	Reverse	Exact
C0B60C80	339	53	Reverse	Exact
C0B60C90	339	60	Reverse	Exact
C0B600A0	339	73	Reverse	Exact
C0B600B0	339	80	Reverse	Exact
C0B600D0	339	93	Reverse	Exact
C0B600F0	339	107	Reverse	Exact
C0B600H0	339	120	Reverse	Exact
C0B600J0	339	133	Reverse	Exact
C0B600N0	339	160	Reverse	Exact
C0B600T0	339	200	Reverse	Exact
C0B600Z0	339	240	Reverse	Exact
C0B60000	339	67	Reverse	Exact
C0B60050	339	33	Reverse	Exact
C0B60060	339	40	Reverse	Exact
C0B60070	339	47	Reverse	Exact
C0B60080	339	53	Reverse	Exact
C0B60090	339	60	Reverse	Exact
C0C055A0	16951	73	Normal	Exact
C0C055B0	16951	80	Normal	Exact
C0C055D0	16951	93	Normal	Exact
C0C055F0	16951	107	Normal	Exact
C0C055H0	16951	120	Normal	Exact
C0C055J0	16951	133	Normal	Exact
C0C055N0	16951	160	Normal	Exact
C0C055T0	16951	200	Normal	Exact
C0C055Z0	16951	240	Normal	Exact
C0C05500	16951	67	Normal	Exact
C0C05560	16951	40	Normal	Exact
C0C05570	16951	47	Normal	Exact
C0C05580	16951	53	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0C05590	16951	60	Normal	Exact
C0C075A0	16971	73	Bold	Exact
C0C075B0	16971	80	Bold	Exact
C0C075D0	16971	93	Bold	Exact
C0C075F0	16971	107	Bold	Exact
C0C075H0	16971	120	Bold	Exact
C0C075J0	16971	133	Bold	Exact
C0C075N0	16971	160	Bold	Exact
C0C075T0	16971	200	Bold	Exact
C0C075Z0	16971	240	Bold	Exact
C0C07500	16971	67	Bold	Exact
C0C07560	16971	40	Bold	Exact
C0C07570	16971	47	Bold	Exact
C0C07580	16971	53	Bold	Exact
C0C07590	16971	60	Bold	Exact
C0C155A0	17079	73	Italic	Exact
C0C155B0	17079	80	Italic	Exact
C0C155D0	17079	93	Italic	Exact
C0C155F0	17079	107	Italic	Exact
C0C155H0	17079	120	Italic	Exact
C0C155J0	17079	133	Italic	Exact
C0C155N0	17079	160	Italic	Exact
C0C155T0	17079	200	Italic	Exact
C0C155Z0	17079	240	Italic	Exact
C0C15500	17079	67	Italic	Exact
C0C15560	17079	40	Italic	Exact
C0C15570	17079	47	Italic	Exact
C0C15580	17079	53	Italic	Exact
C0C15590	17079	60	Italic	Exact
C0C175A0	17099	73	Bold Italic	Exact
C0C175B0	17099	80	Bold Italic	Exact
C0C175D0	17099	93	Bold Italic	Exact
C0C175F0	17099	107	Bold Italic	Exact
C0C175H0	17099	120	Bold Italic	Exact
C0C175J0	17099	133	Bold Italic	Exact
C0C175N0	17099	160	Bold Italic	Exact
C0C175T0	17099	200	Bold Italic	Exact
C0C175Z0	17099	240	Bold Italic	Exact
C0C17500	17099	67	Bold Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0C17560	17099	40	Bold Italic	Exact
C0C17570	17099	47	Bold Italic	Exact
C0C17580	17099	53	Bold Italic	Exact
C0C17590	17099	60	Bold Italic	Exact
C0D0GB10	39	144	Bold	Exact
C0D0GB12	69	120	Bold	Exact
C0D0GC15	231	96	Normal	Not Exact
C0D0GI12	68	120	Italic	Exact
C0D0GL10	303	144	Normal	Not Exact
C0D0GL12	303	120	Normal	Not Exact
C0D0GL15	303	96	Normal	Not Exact
C0D0GP12	174	120	Normal	Exact
C0D0GR10	310	144	Normal	Not Exact
C0D0GT10	40	144	Normal	Exact
C0D0GT12	66	120	Normal	Exact
C0D0GT13	203	108	Normal	Exact
C0D0GT15	230	96	Normal	Exact
C0D0GT18	275	78	Normal	Not Exact
C0D0GT20	281	72	Normal	Exact
C0D0GT24	290	54	Normal	Exact
C0D0RT10	41	144	Normal	Exact
C0D0SB12	72	120	Bold	Exact
C0D0SI10	43	144	Italic	Exact
C0D0SI12	71	120	Italic	Exact
C0D0SO12	332	120	Normal	Not Exact
C0D0ST10	42	144	Normal	Exact
C0D0ST12	70	120	Normal	Exact
C0D0ST15	229	96	Normal	Exact
C0G055A0	4663	73	Normal	Exact
C0G055B0	4663	80	Normal	Exact
C0G055D0	4663	93	Normal	Exact
C0G055F0	4663	107	Normal	Exact
C0G055H0	4663	120	Normal	Exact
C0G055J0	4663	133	Normal	Exact
C0G055N0	4663	160	Normal	Exact
C0G055T0	4663	200	Normal	Exact
C0G055Z0	4663	240	Normal	Exact
C0G05500	4663	67	Normal	Exact
C0G05560	4663	40	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0G05570	4663	47	Normal	Exact
C0G05580	4663	53	Normal	Exact
C0G05590	4663	60	Normal	Exact
C0G075A0	4683	73	Bold	Exact
C0G075B0	4683	80	Bold	Exact
C0G075D0	4683	93	Bold	Exact
C0G075F0	4683	107	Bold	Exact
C0G075H0	4683	120	Bold	Exact
C0G075J0	4683	133	Bold	Exact
C0G075N0	4683	160	Bold	Exact
C0G075T0	4683	200	Bold	Exact
C0G075Z0	4683	240	Bold	Exact
C0G07500	4683	67	Bold	Exact
C0G07560	4683	40	Bold	Exact
C0G07570	4683	47	Bold	Exact
C0G07580	4683	53	Bold	Exact
C0G07590	4683	60	Bold	Exact
C0G155A0	4791	73	Italic	Exact
C0G155B0	4791	80	Italic	Exact
C0G155D0	4791	93	Italic	Exact
C0G155F0	4791	107	Italic	Exact
C0G155H0	4791	120	Italic	Exact
C0G155J0	4791	133	Italic	Exact
C0G155N0	4791	160	Italic	Exact
C0G155T0	4791	200	Italic	Exact
C0G155Z0	4791	240	Italic	Exact
C0G15500	4791	67	Italic	Exact
C0G15560	4791	40	Italic	Exact
C0G15570	4791	47	Italic	Exact
C0G15580	4791	53	Italic	Exact
C0G15590	4791	60	Italic	Exact
C0G175A0	4811	73	Bold Italic	Exact
C0G175B0	4811	80	Bold Italic	Exact
C0G175D0	4811	93	Bold Italic	Exact
C0G175F0	4811	107	Bold Italic	Exact
C0G175H0	4811	120	Bold Italic	Exact
C0G175J0	4811	133	Bold Italic	Exact
C0G175N0	4811	160	Bold Italic	Exact
C0G175T0	4811	200	Bold Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0G175Z0	4811	240	Bold Italic	Exact
C0G17500	4811	67	Bold Italic	Exact
C0G17560	4811	40	Bold Italic	Exact
C0G17570	4811	47	Bold Italic	Exact
C0G17580	4811	53	Bold Italic	Exact
C0G17590	4811	60	Bold Italic	Exact
C0H200A0	2304	73	Normal	Exact
C0H200B0	2304	80	Normal	Exact
C0H200D0	2304	93	Normal	Exact
C0H200F0	2304	107	Normal	Exact
C0H200H0	2304	120	Normal	Exact
C0H200J0	2304	133	Normal	Exact
C0H200N0	2304	160	Normal	Exact
C0H200T0	2304	200	Normal	Exact
C0H200Z0	2304	240	Normal	Exact
C0H20000	2304	67	Normal	Exact
C0H20060	2304	40	Normal	Exact
C0H20070	2304	47	Normal	Exact
C0H20080	2304	53	Normal	Exact
C0H20090	2304	60	Normal	Exact
C0H300A0	2306	73	Italic	Exact
C0H300B0	2306	80	Italic	Exact
C0H300D0	2306	93	Italic	Exact
C0H300F0	2306	107	Italic	Exact
C0H300H0	2306	120	Italic	Exact
C0H300J0	2306	133	Italic	Exact
C0H300N0	2306	160	Italic	Exact
C0H300T0	2306	200	Italic	Exact
C0H300Z0	2306	240	Italic	Exact
C0H30000	2306	67	Italic	Exact
C0H30060	2306	40	Italic	Exact
C0H30070	2306	47	Italic	Exact
C0H30080	2306	53	Italic	Exact
C0H30090	2306	60	Italic	Exact
C0H400A0	2305	73	Bold	Exact
C0H400B0	2305	80	Bold	Exact
C0H400D0	2305	93	Bold	Exact
C0H400F0	2305	107	Bold	Exact
C0H400H0	2305	120	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0H400J0	2305	133	Bold	Exact
C0H400N0	2305	160	Bold	Exact
C0H400T0	2305	200	Bold	Exact
C0H400Z0	2305	240	Bold	Exact
C0H40000	2305	67	Bold	Exact
C0H40060	2305	40	Bold	Exact
C0H40070	2305	47	Bold	Exact
C0H40080	2305	53	Bold	Exact
C0H40090	2305	60	Bold	Exact
C0H500A0	2307	73	Bold Italic	Exact
C0H500B0	2307	80	Bold Italic	Exact
C0H500D0	2307	93	Bold Italic	Exact
C0H500F0	2307	107	Bold Italic	Exact
C0H500H0	2307	120	Bold Italic	Exact
C0H500J0	2307	133	Bold Italic	Exact
C0H500N0	2307	160	Bold Italic	Exact
C0H500T0	2307	200	Bold Italic	Exact
C0H500Z0	2307	240	Bold Italic	Exact
C0H50000	2307	67	Bold Italic	Exact
C0H50060	2307	40	Bold Italic	Exact
C0H50070	2307	47	Bold Italic	Exact
C0H50080	2307	53	Bold Italic	Exact
C0H50090	2307	60	Bold Italic	Exact
C0J055J0	37431	133	Normal	Not Exact
C0J055Z0	37431	240	Normal	Not Exact
C0L0AD10	45	144	Normal	Exact
C0L0AD12	76	120	Normal	Exact
C0L0AG10	45	144	Normal	Not Exact
C0L0AG12	76	120	Normal	Not Exact
C0L0AG15	219	96	Normal	Not Exact
C0L0AI10	58	144	Italic	Not Exact
C0L0AI12	105	120	Italic	Exact
C0L0AT10	45	144	Normal	Not Exact
C0L0AT12	76	120	Normal	Not Exact
C0L0DUMP	230	96	Normal	Not Exact
C0L0FM10	30	144	Normal	Not Exact
C0L0FM12	80	120	Normal	Not Exact
C0L0FM15	225	96	Normal	Not Exact
C0L0GU10	312	144	Normal	Not Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0L0GU12	312	120	Normal	Not Exact
C0L0GU15	312	96	Normal	Not Exact
C0L0KATA	433	144	Normal	Exact
C0L0KN12	433	120	Normal	Exact
C0L0KN20	433	84	Normal	Not Exact
C0L0TU10	334	144	Normal	Not Exact
C0L00AOA	19	144	Normal	Exact
C0L00AON	19	144	Normal	Exact
C0L00APL	45	144	Normal	Not Exact
C0L00BOA	3	144	Normal	Exact
C0L00BON	3	144	Normal	Exact
C0L00GSC	398	96	Normal	Not Exact
C0L00GUC	311	96	Normal	Not Exact
C0L00OAB	3	144	Normal	Exact
C0L00T11	333	144	Normal	Not Exact
C0MO55A0	50231	73	Normal	Not Exact
C0MO55B0	50231	80	Normal	Not Exact
C0MO55B1	50231	320	Normal	Not Exact
C0MO55D0	50231	93	Normal	Not Exact
C0MO55F0	50231	107	Normal	Not Exact
C0MO55H0	50231	120	Normal	Not Exact
C0MO55H1	50231	360	Normal	Not Exact
C0MO55J0	50231	133	Normal	Not Exact
C0MO55L0	50231	147	Normal	Not Exact
C0MO55N0	50231	160	Normal	Not Exact
C0MO55N1	50231	400	Normal	Not Exact
C0MO55R0	50231	187	Normal	Not Exact
C0MO55T0	50231	200	Normal	Not Exact
C0MO55V0	50231	213	Normal	Not Exact
C0MO55Z0	50231	240	Normal	Not Exact
C0MO55Z1	50231	480	Normal	Not Exact
C0MO5500	50231	67	Normal	Not Exact
C0MO5541	50231	267	Normal	Not Exact
C0MO5560	50231	40	Normal	Not Exact
C0MO5570	50231	47	Normal	Not Exact
C0MO5580	50231	53	Normal	Not Exact
C0MO5581	50231	293	Normal	Not Exact
C0MO5590	50231	60	Normal	Not Exact
C0MP55A0	49463	73	Normal	Not Exact

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0MP55B0	49463	80	Normal	Not Exact
C0MP55D0	49463	93	Normal	Not Exact
C0MP55F0	49463	107	Normal	Not Exact
C0MP55H0	49463	120	Normal	Not Exact
C0MP55N0	49463	160	Normal	Not Exact
C0MP55Z0	49463	240	Normal	Not Exact
C0MP5500	49463	67	Normal	Not Exact
C0MP5560	49463	40	Normal	Not Exact
C0MP5570	49463	47	Normal	Not Exact
C0MP5580	49463	53	Normal	Not Exact
C0MP5590	49463	60	Normal	Not Exact
C0MP75A0	49483	73	Bold	Not Exact
C0MP75B0	49483	80	Bold	Not Exact
C0MP75D0	49483	93	Bold	Not Exact
C0MP75F0	49483	107	Bold	Not Exact
C0MP75H0	49483	120	Bold	Not Exact
C0MP75N0	49483	160	Bold	Not Exact
C0MP75Z0	49483	240	Bold	Not Exact
C0MP7500	49483	67	Bold	Not Exact
C0MP7560	49483	40	Bold	Not Exact
C0MP7570	49483	47	Bold	Not Exact
C0MP7580	49483	53	Bold	Not Exact
C0MP7590	49483	60	Bold	Not Exact
C0MQ55A0	49719	73	Normal	Not Exact
C0MQ55B0	49719	80	Normal	Not Exact
C0MQ55D0	49719	93	Normal	Not Exact
C0MQ55F0	49719	107	Normal	Not Exact
C0MQ55H0	49719	120	Normal	Not Exact
C0MQ55N0	49719	160	Normal	Not Exact
C0MQ55Z0	49719	240	Normal	Not Exact
C0MQ5500	49719	67	Normal	Not Exact
C0MQ5560	49719	40	Normal	Not Exact
C0MQ5570	49719	47	Normal	Not Exact
C0MQ5580	49719	53	Normal	Not Exact
C0MQ5590	49719	60	Normal	Not Exact
C0MQ75A0	49739	73	Bold	Not Exact
C0MQ75B0	49739	80	Bold	Not Exact
C0MQ75D0	49739	93	Bold	Not Exact
C0MQ75F0	49739	107	Bold	Not Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0MQ75H0	49739	120	Bold	Not Exact
C0MQ75N0	49739	160	Bold	Not Exact
C0MQ75Z0	49739	240	Bold	Not Exact
C0MQ7500	49739	67	Bold	Not Exact
C0MQ7560	49739	40	Bold	Not Exact
C0MQ7570	49739	47	Bold	Not Exact
C0MQ7580	49739	53	Bold	Not Exact
C0MQ7590	49739	60	Bold	Not Exact
C0N200A0	2308	73	Normal	Exact
C0N200B0	2308	80	Normal	Exact
C0N200D0	2308	93	Normal	Exact
C0N200F0	2308	107	Normal	Exact
C0N200H0	2308	120	Normal	Exact
C0N200J0	2308	133	Normal	Exact
C0N200N0	2308	160	Normal	Exact
C0N200T0	2308	200	Normal	Exact
C0N200Z0	2308	240	Normal	Exact
C0N20000	2308	67	Normal	Exact
C0N20060	2308	40	Normal	Exact
C0N20070	2308	47	Normal	Exact
C0N20080	2308	53	Normal	Exact
C0N20090	2308	60	Normal	Exact
C0N204B0	2308	80	Normal	Not Exact
C0N300A0	2310	73	Italic	Exact
C0N300B0	2310	80	Italic	Exact
C0N300D0	2310	93	Italic	Exact
C0N300F0	2310	107	Italic	Exact
C0N300H0	2310	120	Italic	Exact
C0N300J0	2310	133	Italic	Exact
C0N300N0	2310	160	Italic	Exact
C0N300T0	2310	200	Italic	Exact
C0N300Z0	2310	240	Italic	Exact
C0N30000	2310	67	Italic	Exact
C0N30060	2310	40	Italic	Exact
C0N30070	2310	47	Italic	Exact
C0N30080	2310	53	Italic	Exact
C0N30090	2310	60	Italic	Exact
C0N304B0	2310	80	Italic	Not Exact
C0N400A0	2309	73	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0N400B0	2309	80	Bold	Exact
C0N400D0	2309	93	Bold	Exact
C0N400F0	2309	107	Bold	Exact
C0N400H0	2309	120	Bold	Exact
C0N400J0	2309	133	Bold	Exact
C0N400N0	2309	160	Bold	Exact
C0N400T0	2309	200	Bold	Exact
C0N400Z0	2309	240	Bold	Exact
C0N40000	2309	67	Bold	Exact
C0N40060	2309	40	Bold	Exact
C0N40070	2309	47	Bold	Exact
C0N40080	2309	53	Bold	Exact
C0N40090	2309	60	Bold	Exact
C0N404B0	2309	80	Bold	Not Exact
C0N404H0	2309	120	Bold	Not Exact
C0N404N0	2309	160	Bold	Not Exact
C0N500A0	2311	73	Bold Italic	Exact
C0N500B0	2311	80	Bold Italic	Exact
C0N500D0	2311	93	Bold Italic	Exact
C0N500F0	2311	107	Bold Italic	Exact
C0N500H0	2311	120	Bold Italic	Exact
C0N500J0	2311	133	Bold Italic	Exact
C0N500N0	2311	160	Bold Italic	Exact
C0N500T0	2311	200	Bold Italic	Exact
C0N500Z0	2311	240	Bold Italic	Exact
C0N50000	2311	67	Bold Italic	Exact
C0N50060	2311	40	Bold Italic	Exact
C0N50070	2311	47	Bold Italic	Exact
C0N50080	2311	53	Bold Italic	Exact
C0N50090	2311	60	Bold Italic	Exact
C0N504B0	2311	80	Bold Italic	Not Exact
C0N504H0	2311	120	Bold Italic	Not Exact
C0N504N0	2311	160	Bold Italic	Not Exact
C0OCRA10	19	144	Normal	Exact
C0OCRB10	3	144	Normal	Exact
C0P055B0	49719	80	Normal	Exact
C0P05500	49719	67	Normal	Exact
C0P05560	49719	40	Normal	Exact
C0P05580	49719	53	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0P075B0	49739	80	Bold	Exact
C0P07500	49739	67	Bold	Exact
C0P07560	49739	40	Bold	Exact
C0P07580	49739	53	Bold	Exact
C0Q055B0	49719	80	Normal	Not Exact
C0Q05500	49463	67	Normal	Not Exact
C0Q05560	49463	40	Normal	Not Exact
C0Q05580	49463	53	Normal	Not Exact
C0Q075B0	49483	80	Bold	Not Exact
C0Q07500	49483	67	Bold	Not Exact
C0Q07560	49483	40	Bold	Not Exact
C0Q07580	49483	53	Bold	Not Exact
C0S0AE10	26	144	Normal	Exact
C0S0AE10	45	144	Normal	Exact
C0S0AE20	280	72	Normal	Exact
C0S0AP13	206	108	Normal	Not Exact
C0S0AP20	280	72	Normal	Not Exact
C0S0BITR	155	120	Italic	Exact
C0S0BRTR	159	120	Normal	Exact
C0S0CB10	46	144	Bold	Exact
C0S0CB12	108	120	Bold	Exact
C0S0CB15	214	96	Bold	Exact
C0S0CD15	417	96	Double Wide	Exact
C0S0CE10	11	144	Normal	Not Exact
C0S0CE12	85	120	Normal	Exact
C0S0CH10	37	144	Normal	Not Exact
C0S0CI10	18	144	Italic	Exact
C0S0CI12	92	120	Italic	Exact
C0S0CI15	215	96	Italic	Exact
C0S0CO10	302	144	Normal	Not Exact
C0S0CR10	11	144	Normal	Exact
C0S0CR12	85	120	Normal	Exact
C0S0CR15	223	96	Normal	Exact
C0S0CW15	425	96	Double Wide Italic	Exact
C0S0DOTR	175	120	Normal	Exact
C0S0D224	203	108	Normal	Exact
C0S0D225	203	108	Normal	Exact
C0S0D226	201	108	Bold	Exact

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0S0D227	202	108	Italic	Exact
C0S0EBTR	163	120	Bold	Exact
C0S0EITR	162	120	Italic	Exact
C0S0ELTR	173	120	Normal	Exact
C0S0EOTR	196	120	Normal	Not Exact
C0S0ESTR	160	120	Normal	Exact
C0S0LB12	110	120	Bold	Exact
C0S0LR12	87	120	Normal	Exact
C0S0OB10	38	144	Bold	Exact
C0S0OR10	5	144	Normal	Exact
C0S0PB12	111	120	Bold	Exact
C0S0PI12	112	120	Italic	Exact
C0S0PR10	12	144	Normal	Exact
C0S0PR12	86	120	Normal	Exact
C0S0SR12	84	120	Normal	Exact
C0S0SYM0	49975	67	Normal	Not Exact
C0S0SYM2	49975	80	Normal	Not Exact
C0S0S192	80	120	Normal	Not Exact
C0S0S193	80	120	Normal	Not Exact
C0S0S198	30	144	Normal	Not Exact
C0S055A0	28983	73	Normal	Exact
C0S055B0	28983	80	Normal	Exact
C0S055D0	28983	93	Normal	Exact
C0S055F0	28983	107	Normal	Exact
C0S055H0	28983	120	Normal	Exact
C0S055J0	28983	133	Normal	Exact
C0S055N0	28983	160	Normal	Exact
C0S055T0	28983	200	Normal	Exact
C0S055Z0	28983	240	Normal	Exact
C0S05500	28983	67	Normal	Exact
C0S05560	28983	40	Normal	Exact
C0S05570	28983	47	Normal	Exact
C0S05580	28983	53	Normal	Exact
C0S05590	28983	60	Normal	Exact
C0S075A0	29003	73	Bold	Exact
C0S075B0	29003	80	Bold	Exact
C0S075D0	29003	93	Bold	Exact
C0S075F0	29003	107	Bold	Exact
C0S075H0	29003	120	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0S075J0	29003	133	Bold	Exact
C0S075N0	29003	160	Bold	Exact
C0S075T0	29003	200	Bold	Exact
C0S075Z0	29003	240	Bold	Exact
C0S07500	29003	67	Bold	Exact
C0S07560	29003	40	Bold	Exact
C0S07570	29003	47	Bold	Exact
C0S07580	29003	53	Bold	Exact
C0S07590	29003	60	Bold	Exact
C0S155A0	29111	73	Italic	Exact
C0S155B0	29111	80	Italic	Exact
C0S155D0	29111	93	Italic	Exact
C0S155F0	29111	107	Italic	Exact
C0S155H0	29111	120	Italic	Exact
C0S155J0	29111	133	Italic	Exact
C0S155N0	29111	160	Italic	Exact
C0S155T0	29111	200	Italic	Exact
C0S155Z0	29111	240	Italic	Exact
C0S15500	29111	67	Italic	Exact
C0S15560	29111	40	Italic	Exact
C0S15570	29111	47	Italic	Exact
C0S15580	29111	53	Italic	Exact
C0S15590	29111	60	Italic	Exact
C0S175A0	29131	73	Bold Italic	Exact
C0S175B0	29131	80	Bold Italic	Exact
C0S175D0	29131	93	Bold Italic	Exact
C0S175F0	29131	107	Bold Italic	Exact
C0S175H0	29131	120	Bold Italic	Exact
C0S175J0	29131	133	Bold Italic	Exact
C0S175N0	29131	160	Bold Italic	Exact
C0S175T0	29131	200	Bold Italic	Exact
C0S175Z0	29131	240	Bold Italic	Exact
C0S17500	29131	67	Bold Italic	Exact
C0S17560	29131	40	Bold Italic	Exact
C0S17570	29131	47	Bold Italic	Exact
C0S17580	29131	53	Bold Italic	Exact
C0S17590	29131	60	Bold Italic	Exact
C0T055A0	4407	73	Normal	Exact
C0T055B0	4407	80	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0T055B1	4407	320	Normal	Exact
C0T055D0	4407	93	Normal	Exact
C0T055F0	4407	107	Normal	Exact
C0T055H0	4407	120	Normal	Exact
C0T055J0	4407	133	Normal	Exact
C0T055N0	4407	160	Normal	Exact
C0T055N1	4407	400	Normal	Exact
C0T055T0	4407	200	Normal	Exact
C0T055Z0	4407	240	Normal	Exact
C0T055Z1	4407	480	Normal	Exact
C0T05500	4407	67	Normal	Exact
C0T05560	4407	40	Normal	Exact
C0T05570	4407	47	Normal	Exact
C0T05580	4407	53	Normal	Exact
C0T05590	4407	60	Normal	Exact
C0T075A0	4427	73	Bold	Exact
C0T075B0	4427	80	Bold	Exact
C0T075B1	4427	320	Bold	Exact
C0T075D0	4427	93	Bold	Exact
C0T075F0	4427	107	Bold	Exact
C0T075H0	4427	120	Bold	Exact
C0T075J0	4427	133	Bold	Exact
C0T075N0	4427	160	Bold	Exact
C0T075N1	4427	400	Bold	Exact
C0T075T0	4427	200	Bold	Exact
C0T075Z0	4427	240	Bold	Exact
C0T075Z1	4427	480	Bold	Exact
C0T07500	4427	67	Bold	Exact
C0T07560	4427	40	Bold	Exact
C0T07570	4427	47	Bold	Exact
C0T07580	4427	53	Bold	Exact
C0T07590	4427	60	Bold	Exact
C0T155A0	4535	73	Italic	Exact
C0T155B0	4535	80	Italic	Exact
C0T155B1	4535	320	Italic	Exact
C0T155D0	4535	93	Italic	Exact
C0T155F0	4535	107	Italic	Exact
C0T155H0	4535	120	Italic	Exact
C0T155J0	4535	133	Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0T155N0	4535	160	Italic	Exact
C0T155N1	4535	400	Italic	Exact
C0T155T0	4535	200	Italic	Exact
C0T155Z0	4535	240	Italic	Exact
C0T155Z1	4535	480	Italic	Exact
C0T15500	4535	67	Italic	Exact
C0T15560	4535	40	Italic	Exact
C0T15570	4535	47	Italic	Exact
C0T15580	4535	53	Italic	Exact
C0T15590	4535	60	Italic	Exact
C0T175A0	4555	73	Bold Italic	Exact
C0T175B0	4555	80	Bold Italic	Exact
C0T175B1	4555	320	Bold Italic	Exact
C0T175D0	4555	93	Bold Italic	Exact
C0T175F0	4555	107	Bold Italic	Exact
C0T175H0	4555	120	Bold Italic	Exact
C0T175J0	4555	133	Bold Italic	Exact
C0T175N0	4555	160	Bold Italic	Exact
C0T175N1	4555	400	Bold Italic	Exact
C0T175T0	4555	200	Bold Italic	Exact
C0T175Z0	4555	240	Bold Italic	Exact
C0T175Z1	4555	480	Bold Italic	Exact
C0T17500	4555	67	Bold Italic	Exact
C0T17560	4555	40	Bold Italic	Exact
C0T17570	4555	47	Bold Italic	Exact
C0T17580	4555	53	Bold Italic	Exact
C0T17590	4555	60	Bold Italic	Exact
C0V055A0	33847	73	Normal	Exact
C0V055B0	33847	80	Normal	Exact
C0V055D0	33847	93	Normal	Exact
C0V055F0	33847	107	Normal	Exact
C0V055H0	33847	120	Normal	Exact
C0V055J0	33847	133	Normal	Exact
C0V055N0	33847	160	Normal	Exact
C0V055T0	33847	200	Normal	Exact
C0V055Z0	33847	240	Normal	Exact
C0V05500	33847	67	Normal	Exact
C0V05560	33847	40	Normal	Exact
C0V05570	33847	47	Normal	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0V05580	33847	53	Normal	Exact
C0V05590	33847	60	Normal	Exact
C0V075A0	33867	73	Bold	Exact
C0V075B0	33867	80	Bold	Exact
C0V075D0	33867	93	Bold	Exact
C0V075F0	33867	107	Bold	Exact
C0V075H0	33867	120	Bold	Exact
C0V075J0	33867	133	Bold	Exact
C0V075N0	33867	160	Bold	Exact
C0V075T0	33867	200	Bold	Exact
C0V075Z0	33867	240	Bold	Exact
C0V07500	33867	67	Bold	Exact
C0V07560	33867	40	Bold	Exact
C0V07570	33867	47	Bold	Exact
C0V07580	33867	53	Bold	Exact
C0V07590	33867	60	Bold	Exact
C0V155A0	33975	73	Italic	Exact
C0V155B0	33975	80	Italic	Exact
C0V155D0	33975	93	Italic	Exact
C0V155F0	33975	107	Italic	Exact
C0V155H0	33975	120	Italic	Exact
C0V155J0	33975	133	Italic	Exact
C0V155N0	33975	160	Italic	Exact
C0V155T0	33975	200	Italic	Exact
C0V155Z0	33975	240	Italic	Exact
C0V15500	33975	67	Italic	Exact
C0V15560	33975	40	Italic	Exact
C0V15570	33975	47	Italic	Exact
C0V15580	33975	53	Italic	Exact
C0V15590	33975	60	Italic	Exact
C0V175A0	33995	73	Bold Italic	Exact
C0V175B0	33995	80	Bold Italic	Exact
C0V175D0	33995	93	Bold Italic	Exact
C0V175F0	33995	107	Bold Italic	Exact
C0V175H0	33995	120	Bold Italic	Exact
C0V175J0	33995	133	Bold Italic	Exact
C0V175N0	33995	160	Bold Italic	Exact
C0V175T0	33995	200	Bold Italic	Exact
C0V175Z0	33995	240	Bold Italic	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0V17500	33995	67	Bold Italic	Exact
C0V17560	33995	40	Bold Italic	Exact
C0V17570	33995	47	Bold Italic	Exact
C0V17580	33995	53	Bold Italic	Exact
C0V17590	33995	60	Bold Italic	Exact
C0Z05640	33080	27	Normal	Not Exact
C04200B0	416	144	Normal	Exact
C04200D0	416	168	Normal	Exact
C04200J0	416	240	Normal	Exact
C0420000	416	120	Normal	Exact
C0420070	416	84	Normal	Exact
C0420080	416	96	Normal	Exact
C04202B0	416	144	Normal	Exact
C0420200	416	120	Normal	Exact
C0420270	416	84	Normal	Exact
C0420280	416	96	Normal	Exact
C04203B0	416	144	Normal	Exact
C0420300	416	120	Normal	Exact
C0420380	416	96	Normal	Exact
C04204B0	416	144	Normal	Exact
C0420400	416	120	Normal	Exact
C0420480	416	96	Normal	Not Exact
C04205B0	49	416	Normal	Exact
C0420500	416	120	Normal	Exact
C0420570	416	84	Normal	Not Exact
C0420580	416	96	Normal	Not Exact
C04300B0	424	144	Italic	Exact
C04300D0	424	168	Italic	Exact
C04300J0	424	240	Italic	Exact
C0430000	424	120	Italic	Exact
C0430070	424	84	Italic	Exact
C0430080	424	96	Italic	Exact
C04304B0	424	144	Italic	Exact
C0430400	424	120	Italic	Exact
C04400B0	420	144	Bold	Exact
C04400D0	420	168	Bold	Exact
C04400J0	420	240	Bold	Exact
C0440000	420	120	Bold	Exact
C0440070	420	84	Bold	Exact

Table 39. Host Resident to Printer Resident Font Character Set Mapping (continued)

Font Character Set Name	Registered Font ID	Font Width	Font Attributes	Map Fidelity
C0440080	420	96	Bold	Exact
C0440200	420	120	Bold	Not Exact
C0440300	420	120	Bold	Not Exact
C04404B0	420	144	Bold	Exact
C04404D0	420	168	Bold	Not Exact
C0440470	420	84	Bold	Exact
C04405B0	420	144	Bold	Exact
C0440500	420	120	Bold	Not Exact
C04500B0	428	144	Bold Italic	Exact
C04500D0	428	168	Bold Italic	Exact
C04500J0	428	240	Bold Italic	Exact
C0450000	428	120	Bold Italic	Exact
C0450070	428	84	Bold Italic	Exact
C0450080	428	96	Bold Italic	Exact
C0450300	428	120	Bold Italic	Not Exact
C04504B0	428	144	Bold Italic	Exact
C04504D0	428	168	Bold Italic	Not Exact

Printer Resident to Host Resident Font Character Set Mapping

Often, when a spooled file is created on OS/400, a font global identifier (FGID) or font ID is specified to be used when the spooled file is printed.

In the case of the 3820, 3825, 3827, 3828, 3829, 3831, 3835, and 3900 printer font IDs are not supported. Therefore, when the spooled file is directed to these printers, the system substitutes font character sets that allow the spooled file to print on these printers.

The following table can help you determine what host resident character sets (font character sets stored on the system) are downloaded to a 3820, 3825, 3827, 3828, 3829, 3831, 3835, or 3900 printer when your spooled file references a registered font identifier (font ID) instead of a host resident font character set.

In the case of *CONTENT fidelity, depending upon the registered font ID value, the font width value, and the font attributes that are requested for a particular font reference, the appropriate host resident font character set is selected to match (as closely as possible) your font request.

In the case of *ABSOLUTE fidelity, depending upon the registered font ID value, the font width value, and the font attributes that are requested for a particular font reference, the appropriate host resident font character set is selected to match exactly your font request. The system also ensures, for *ABSOLUTE fidelity, that the character set is compatible with the code page when mapping from printer resident fonts to host resident fonts.

If the FGID has an asterisk next to it, the appropriate CHRID must be specified to match this entry in the font table regardless of what fidelity value has been specified.

The font width specifies the width of the blank character in 1440ths of an inch. This is an indicator of how many characters fit per inch of space on the paper.

Some FGIDs, such as 416, have multiple widths associated with them. The **Font Width** column of the table is blank for these FGIDs. Also, the primary and secondary character set names of these FGIDs have an XX in the last two positions, which identify the size of the font. For uniformly spaced fonts, there are six widths: 84, 96, 120, 144, 168, and 240. For typographic fonts, there are 14 widths: 40, 47, 53, 60, 67, 73, 80, 93, 107, 120, 133, 160, 200 and 240. The system determines whether a uniformly spaced font or a typographic font is needed and then selects the host resident character set name based on the width provided.

The first choice is used if it is present on your iSeries server. The second choice is used if the first choice cannot be found. The Map Fidelity column indicates whether or not the first choice is considered to be an exact match to the printer resident font (font ID) that is requested in your spooled file. As a rule, the second choice is not considered to be an exact match.

If the first choice contains a metric-only font character set name, the system uses the second choice regardless of the fidelity setting. A metric-only font character set name begins with the characters *C0E*.

If code page (CPGID) 259 (the symbol font code page) has been specified, this table is not used. Instead, if the FGID specified is a 10-pitch font, the C0S0SYM2 character set is substituted; otherwise, if the FGID specified is anything other than a 10-pitch font, the C0S0SYM0 character set is substituted.

Table 40. Printer Resident to Host Resident Font Character Set Mapping

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
2	144	Normal	Exact	C0E0DE10	C0S0CR12
2	144	Normal	Exact	C0E0DE0R	C0S0SYM0
3	144	Normal	Exact	C0L00BOA	
3	144	Normal	Exact	C0S0SYM2	
5	144	Normal	Exact	C0S0OR10	
5	144	Bold	Exact	C0S0OB10	
5	144	Normal	Exact	C0E0OR10	C0S0OR10
5	144	Normal	Exact	C0S0SYM2	
10	144	Normal	Exact	C0E0CY10	C04203B0
10	144	Normal	Exact	C0E0CY0R	C0S0SYM0
11	144	Normal	Exact	C0S0CR10	
11	144	Bold	Exact	C0S0CB10	
11	144	Italic	Exact	C0S0CI10	
11	144	Normal	Exact	C0S0SYM2	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
11	144	Normal	Exact	C0E0CR0K	C04203B0
11	144	Normal	Exact	C0E0CR0F	C0S0CR10
11	144	Normal	Exact	C0E0CR0N	C04203B0
11	144	Normal	Exact	C0E0CR0Q	C04203B0
11	144	Normal	Exact	C0E0CR0G	C04202B0
11	144	Normal	Exact	C0E0CR0H	C04202B0
12	144	Normal	Exact	C0S0PR10	
12	144	Normal	Exact	C0S0SYM2	
12	144	Normal	Exact	C0E0PR0G	C04202B0
12	144	Normal	Exact	C0E0PR0H	C04202B0
13	144	Normal	Not Exact	C0S0CR10	
13	144	Normal	Exact	C0S0SYM2	
13	144	Bold	Not Exact	C0S0CB10	
18	144	Italic	Exact	C0S0CI10	
18	144	Normal	Exact	C0S0SYM2	
19	144	Normal	Exact	C0L00AOA	
19	144	Normal	Exact	C0S0SYM2	
20	144	Normal	Not Exact	C0S0CR10	
20	144	Bold	Not Exact	C0S0CB10	
20	144	Normal	Exact	C0S0SYM2	
21	144	Normal	Exact	C0E0KA10	C0L0KATA
25	144	Normal	Exact	C0E0PS10	C0S0CR10
26	144	Normal	Exact	C0L0KATA	
26	144	Normal	Exact	C0S0AE10	
26	144	Normal	Exact	C0D0GT10	
26	144	Bold	Exact	C0D0GB10	
30	144	Normal	Exact	C0S0S198	
30	144	Normal	Exact	C0S0SYM2	
36	144	Normal	Exact	C0E0LR10	C0S0CR10
38	144	Bold	Exact	C0S0OB10	
38	144	Bold	Exact	C0S0SYM2	
39	144	Bold	Exact	C0D0GB10	
39	144	Bold	Exact	C0S0SYM2	
40	144	Normal	Exact	C0D0GT10	
40	144	Normal	Exact	C0S0SYM2	
41	144	Normal	Exact	C0D0RT10	
41	144	Normal	Exact	C0S0SYM2	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
42	144	Normal	Exact	C0D0ST10	
42	144	Normal	Exact	C0S0SYM2	
43	144	Italic	Exact	C0D0SI10	
43	144	Italic	Exact	C0S0SYM2	
44	144	Normal	Exact	C0L0KATA	
44	144	Normal	Exact	C0S0SYM2	
45	144	Normal	Exact	C0S0AE10	
46	144	Bold	Exact	C0S0CB10	
46	144	Normal	Exact	C0S0SYM2	
49	144	Normal	Exact	C0E0HR10	C04205B0
50	144	Bold	Exact	C0H0HB10	C04405B0
50	144	Bold	Exact	C0E0HB10	C04405B0
50	144	Normal	Exact	C0S0SYM2	
51	144	Normal	Not Exact	C0S0CR10	
51	144	Bold	Not Exact	C0S0CB10	
52	144	Normal	Not Exact	C0S0CR10	
52	144	Bold	Not Exact	C0S0CB10	
61	144	Normal	Exact	C0E0NR10	C04204B0
62	144	Italic	Exact	C0E0NI10	C04304B0
63	144	Bold	Exact	C0E0NB10	C04404B0
64	144	Bold Italic	Exact	C0E0NM10	C04504B0
66	120	Normal	Exact	C0D0GT12	
66	120	Bold	Exact	C0D0GB12	
66	120	Normal	Exact	C0S0SYM0	
68	120	Italic	Exact	C0D0GI12	
68	120	Italic	Exact	C0S0SYM0	
69	120	Bold	Exact	C0D0GB12	
69	120	Bold	Exact	C0S0SYM0	
70	120	Normal	Exact	C0D0ST12	
70	120	Normal	Exact	C0S0SYM0	
71	120	Italic	Exact	C0D0SI12	
71	120	Italic	Exact	C0S0SYM0	
72	120	Bold	Exact	C0D0SB12	
72	120	Bold	Exact	C0S0SYM0	
74	120	Normal	Not Exact	C0S0CR12	
74	120	Bold	Not Exact	C0S0CB12	
75	120	Normal	Not Exact	C0S0CR12	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
75	120	Bold	Not Exact	C0S0CB12	
76	120	Normal	Exact	C0E0AP12	C0S0AE10
78	120	Normal	Exact	C0E0KA12	C0L0KN12
80	120	Normal	Not Exact	C0S0CR12	
80	120	Bold	Not Exact	C0S0CB12	
80	120	Normal	Exact	C0S0SYM0	
84	120	Normal	Exact	C0S0SR12	
84	120	Normal	Exact	C0E0SR12	C0S0SR12
84	120	Normal	Exact	C0S0SYM0	
85	120	Normal	Exact	C0S0CR12	
85	120	Normal	Exact	C0S0CE12	
85	120	Bold	Exact	C0S0CB12	
85	120	Italic	Exact	C0S0CI12	
85	120	Normal	Exact	C0S0SYM0	
85	120	Normal	Exact	C0E0CREF	C0420200
85	120	Normal	Exact	C0E0CREQ	C0420300
85	120	Normal	Exact	C0E0CREG	C0420200
85	120	Normal	Exact	C0E0CREH	C0420200
86	120	Normal	Exact	C0S0PR12	
86	120	Bold	Exact	C0S0PB12	
86	120	Normal	Exact	C0S0SYM0	
86	120	Normal	Exact	C0E0PREF	C0420200
86	120	Normal	Exact	C0E0PREQ	C0420300
86	120	Normal	Exact	C0E0PREG	C0420200
86	120	Normal	Exact	C0E0PREH	C0420200
87	120	Normal	Exact	C0S0LR12	
87	120	Bold	Exact	C0S0LB12	
87	120	Italic	Exact	C0D0GI12	
87	120	Normal	Exact	C0S0SYM0	
87	120	Normal	Exact	C0E0LRSR	C0S0LR12
87	120	Normal	Exact	C0E0LREK	C0420300
87	120	Normal	Exact	C0E0LREF	C0420200
87	120	Normal	Exact	C0E0LREN	C0420300
87	120	Normal	Exact	C0E0LREQ	C0420300
87	120	Normal	Exact	C0E0LREG	C0420200
87	120	Normal	Exact	C0E0LREH	C0420200
91	120	Italic	Not Exact	C0S0CR12	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
91	120	Bold Italic	Not Exact	C0S0CB12	
91	120	Italic	Exact	C0S0SYM0	
92	120	Italic	Exact	C0S0CI12	
92	120	Normal	Exact	C0E0CIER	C0S0SYM2
95	120	Normal	Exact	C0E0AJ12	C0S0CR12
95	120	Normal	Exact	C0E0AJER	C0S0SYM2
96	120	Bold	Exact	C0E0WB12	C0S0CR12
98	120	Normal	Exact	C0E0HR12	C0420500
103	120	Normal	Exact	C0E0NR12	C0420400
104	120	Italic	Exact	C0E0NI12	C0430400
108	120	Bold	Exact	C0S0CB12	
108	120	Normal	Exact	C0E0CBER	C0S0SYM2
109	120	Italic	Exact	C0E0LI12	C0S0CR12
109	120	Normal	Exact	C0E0LISR	C0S0SYM2
110	120	Bold	Exact	C0S0LB12	
110	120	Bold	Exact	C0E0LBEK	C0440300
110	120	Bold	Exact	C0E0LBEN	C0440300
110	120	Normal	Exact	C0S0SYM0	
111	120	Bold	Exact	C0S0PB12	
111	120	Normal	Exact	C0S0SYM0	
112	120	Italic	Exact	C0S0PI12	
112	120	Normal	Exact	C0S0SYM0	
154	120	Normal	Exact	C0S0ESTR	
154	120	Bold	Exact	C0S0EBTR	
155	120	Bold Italic	Exact	C0S0BITR	
155	120	Bold Italic	Exact	C0E0BIRK	C0450300
155	120	Bold Italic	Exact	C0E0BIRN	C0450300
155	120	Bold	Exact	C0S0SYM0	
157	120	Bold	Exact	C0E0TBTR	C0S0ESTR
157	120	Bold	Exact	C0E0TBRF	C0420200
157	120	Normal	Exact	C0E0TBRR	C0S0SYM2
158	120	Normal	Not Exact	C0S0ESTR	
158	120	Bold	Not Exact	C0S0EBTR	
158	120	Normal	Exact	C0S0SYM0	
159	120	Bold	Exact	C0S0BRTR	
159	120	Bold	Exact	C0E0BRRK	C0440300
159	120	Bold	Exact	C0E0BRRN	C0440300

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
159	120	Bold	Exact	C0E0BRRQ	C0440300
159	120	Bold	Exact	C0E0BRRG	C0440200
159	120	Bold	Exact	C0E0BRRH	C0440200
159	120	Bold	Exact	C0S0SYM0	
160	120	Normal	Exact	C0S0ESTR	
160	120	Bold	Exact	C0S0EBTR	
160	120	Italic	Exact	C0S0EITR	
160	120	Normal	Exact	C0S0SYM0	
162	120	Italic	Exact	C0S0EITR	
162	120	Normal	Exact	C0S0SYM0	
163	120	Bold	Exact	C0S0EBTR	
163	120	Bold	Exact	C0S0SYM0	
164	120	Normal	Exact	C0E0PRTR	C0S0ESTR
164	120	Normal	Exact	C0E0PRRR	C0S0SYM2
167	120	Bold	Normal	C0E0BKTR	C0440500
168	120	Bold	Exact	C0H0BRK2	
168	120	Bold	Exact	C0S0SYM0	
173	120	Normal	Exact	C0S0ELTR	
173	120	Normal	Exact	C0S0SYM0	
174	120	Normal	Exact	C0D0GP12	
175	120	Normal	Exact	C0S0DOTR	
175	120	Normal	Exact	C0S0SYM0	
186	120	Normal	Exact	C0E0RRTR	C0S0ESTR
186	120	Normal	Exact	C0E0RRRR	C0S0SYM2
187	120	Bold	Exact	C0E0RBTR	C0S0ESTR
187	120	Normal	Exact	C0E0RBRR	C0S0SYM2
188	120	Italic	Exact	C0E0RITR	C0S0ESTR
188	120	Normal	Exact	C0E0RIRR	C0S0SYM2
189	120	Bold Italic	Exact	C0E0RMTR	C0S0ESTR
189	120	Normal	Exact	C0E0RMRR	C0S0SYM2
190	120	Normal	Exact	C0E0FRTR	C0S0ESTR
190	120	Normal	Exact	C0E0FRRR	C0S0SYM2
191	120	Bold	Exact	C0E0FBTR	C0S0ESTR
191	120	Normal	Exact	C0E0FBRR	C0S0SYM2
194	120	Italic	Exact	C0E0FITR	C0S0ESTR
194	120	Normal	Exact	C0E0FIRR	C0S0SYM2
195	120	Bold Italic	Exact	C0E0FMTR	C0S0ESTR

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
195	120	Normal	Exact	C0E0FMRR	C0S0SYM2
201	108	Bold	Exact	C0S0D226	C0D0GT13
202	108	Italic	Exact	C0S0D227	C0D0GT13
203	108	Normal	Exact	C0S0D224	C0D0GT13
203	108	Normal	Exact	C0S0D225	C0D0GT13
204	108	Normal	Exact	C0S0D224	C0D0GT13
204	108	Normal	Exact	C0S0D225	C0D0GT13
204	108	Bold	Exact	C0S0D226	C0D0GT13
204	108	Italic	Exact	C0S0D227	C0D0GT13
204	96	Normal	Not Exact	C0S0CR15	
204	96	Bold	Not Exact	C0S0CB15	
204	111	Normal	Exact	C0S0SYM0	
205	96	Normal	Not Exact	C0S0CR15	C0D0GT13
205	96	Bold	Not Exact	C0S0CB15	C0D0GT13
213	96	Normal	Exact	C0E0NR15	C0420480
214	96	Bold	Exact	C0S0CB15	
215	96	Italic	Exact	C0S0CI15	
217	96	Double Wide	Exact	C0S0CD15	
218	96	Double Wide Italic	Exact	C0S0CW15	
221	96	Normal	Not Exact	C0S0CR15	
221	96	Bold	Not Exact	C0S0CB15	
221	96	Normal	Exact	C0E0PR15	C0S0CR15
221	96	Normal	Exact	C0S0SYM0	
222	96	Normal	Exact	C0D0GT15	
222	96	Bold	Not Exact	C0S0CB15	
222	96	Normal	Exact	C0E0LR15	C0S0LR15
222	96	Normal	Exact	C0E0LR5R	C0S0SYM2
223	96	Normal	Exact	C0S0CR15	
223	96	Double Wide	Exact	C0S0CD15	
223	96	Bold	Exact	C0S0CB15	
223	96	Italic	Exact	C0S0CI15	
223	96	Double Wide Italic	Exact	C0S0CW15	
223	96	Normal	Exact	C0E0CR15	C0S0CR15
223	96	Normal	Exact	C0S0SYM0	
223	96	Normal	Exact	C0E0CR5K	C0420380
223	96	Normal	Exact	C0E0CR5N	C0420380

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
223	96	Normal	Exact	C0E0CR5G	C0420280
223	96	Normal	Exact	C0E0CR5H	C0420280
225	96	Normal	Not Exact	C0S0CR15	
225	96	Bold	Not Exact	C0S0CB15	
225	96	Normal	Exact	C0S0SYM0	
226	96	Normal	Exact	C0E0HR15	C0420580
229	96	Normal	Exact	C0D0ST15	C0S0CR15
229	96	Normal	Exact	C0S0SYM0	
230	96	Normal	Exact	C0D0GT15	C0S0CR15
230	96	Normal	Exact	C0S0SYM0	
232	96	Normal	Not Exact	C0S0CR15	
232	96	Bold	Not Exact	C0S0CB15	
233	96	Normal	Exact	C0S0CD15	
244	288	Normal	Exact	C0S0SYM0	
245	144	Normal	Not Exact	C0S0CR10	
245	144	Bold	Not Exact	C0S0CB10	
245	288	Bold	Exact	C0S0SYM2	
248	84	Normal	Exact	C0420580	
249	84	Normal	Exact	C0E0KA17	C0L0KN20
252	78	Normal	Not Exact	C0D0GT18	
252	84	Normal	Exact	C0D0GT18	
252	84	Normal	Exact	C0S0SYM0	
253	84	Bold	Not Exact	C0D0GT18	
253	84	Bold	Exact	C0S0SYM0	
254	78	Normal	Not Exact	C0D0GT18	
254	84	Normal	Exact	C0E0CR7F	C0D0GT18
254	84	Normal	Exact	C0E0CR7G	C0420270
254	84	Normal	Exact	C0E0CR7H	C0420270
254	84	Normal	Exact	C0E0CR17	C0D0GT18
254	84	Normal	Exact	C0S0SYM0	
255	84	Normal	Exact	C0E0LR17	C0D0GT18
255	84	Normal	Exact	C0E0LR7R	C0S0SYM2
256	84	Normal	Exact	C0E0PR17	C0D0GT18
256	84	Normal	Exact	C0E0PR7R	C0S0SYM2
258	78	Normal	Exact	C0D0GT18	
259	78	Normal	Not Exact	C0D0GT18	
266	177	Bold	Exact	C0E0NB08	C04404D0

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
267	177	Bold Italic	Exact	C0E0NM08	C04504D0
275	78	Normal	Exact	C0D0GT18	
279	84	Bold	Exact	C0E0NR17	C0440470
280	72	Normal	Exact	C0S0AE20	
281	72	Normal	Exact	C0D0GT20	
281	72	Normal	Exact	C0E0LR20	C0D0GT20
281	72	Normal	Exact	C0S0SYM0	
282	72	Normal	Exact	C0E0LV20	C0420570
283	72	Bold	Exact	C0E0GN20	C0440470
285	58	Normal	Exact	C0E0LR25	C0D0GT20
290	54	Normal	Exact	C0D0GT24	
290	53	Normal	Not Exact	C0D0GT24	
290	53	Normal	Exact	C0S0SYM0	
300	54	Normal	Exact	C0D0GT18	
304*	54	Normal	Exact	C0620050	C0D0GT20
304*	72	Normal	Exact	C0620060	C0D0GT20
304*	84	Normal	Exact	C0620070	C0D0GT18
304*	96	Normal	Exact	C0620080	C0D0GT15
304*	108	Normal	Exact	C0620090	C0D0GT13
304*	120	Normal	Exact	C0620000	C0D0GT12
304*	144	Normal	Exact	C06200B0	C0D0GT10
304*	168	Normal	Exact	C06200D0	C0D0GT10
304*	240	Normal	Exact	C06200N0	C0D0GT10
304*	72	Normal	Exact	C0620860	C0L0KN20
304*	84	Normal	Exact	C0620870	C0L0KN20
304*	96	Normal	Exact	C0620880	C0L0KN20
304*	120	Normal	Exact	C06208B0	C0L0KATA
304*	168	Normal	Exact	C06208D0	C0L0KATA
304*	240	Normal	Exact	C06208J0	C0L0KATA
305*	144	Normal	Exact	C0920AB0	C0L00AOA
306*	144	Normal	Exact	C0920BB0	C0L00BOA
307*	144	Normal	Exact	C0420P00	C050AE10
318*		Bold	Exact	C07400XX	C050CBXX
319*		Italic	Exact	C07300XX	C050CIXX
322*	144	Normal	Exact	C0440P00	C050AE10
323*	120	Normal	Exact	C0BPOSA0	
323*	144	Normal	Exact	C0BPOS91	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
323*	168	Normal	Exact	C0BPOSB0	
323*	240	Normal	Exact	C0BPOSBN	
326	96	Bold	Exact	C0T40680	
326	120	Bold	Exact	C0T40600	
326	144	Bold	Exact	C0T406B0	
326	180	Bold	Exact	C0T406E0	
327	96	Italic	Exact	C0T30680	
327	120	Italic	Exact	C0T30600	
327	144	Italic	Exact	C0T306B0	
327	180	Italic	Exact	C0T306E0	
328	96	Normal	Exact	C0T20680	
328	120	Normal	Exact	C0T20600	
328	144	Normal	Exact	C0T206B0	
328	180	Normal	Exact	C0T206E0	
335		Normal	Exact	C0B200XX	
335*		Normal	Exact	C0B20CXX	
336		Bold	Exact	C0B400XX	
336*		Bold	Exact	C0B40CXX	
337		Italic	Exact	C0B300XX	
337*		Italic	Exact	C0B30CXX	
338		Bold Italic	Exact	C0B500XX	
338*		Bold Italic	Exact	C0B50CXX	
339		Reverse	Exact	C0B600XX	
339*		Reverse	Exact	C0B60CXX	
400	80	Normal	Exact	C0D0GT18	
400*		Normal	Exact	C05200XX	C0D0GTXX
404*		Normal	Exact	C05400XX	C0D0GTXX
416		Normal	Exact	C04200XX	C0S0CRXX
416	115	Normal	Exact	C0420000	C0S0CR15
416		Normal	Exact	C04202XX	
416		Normal	Exact	C04203XX	
416		Normal	Exact	C04204XX	
416		Normal	Exact	C04205XX	
416*		Normal	Exact	C04201XX	
416*		Normal	Exact	C04207XX	
417	96	Double Wide	Exact	C0S0CD15	
420		Bold	Exact	C04400XX	C0S0CBXX

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
420	115	Normal	Exact	C0440000	C0S0CB15
420		Bold	Exact	C04402XX	
420		Bold	Exact	C04403XX	
420		Bold	Exact	C04404XX	
420		Bold	Exact	C04405XX	
420*		Bold	Exact	C04401XX	
420*		Bold	Exact	C04407XX	
424		Italic	Exact	C04300XX	C0S0CIXX
424	115	Italic	Exact	C0430000	C0S0CI15
424		Italic	Exact	C04302XX	
424		Italic	Exact	C04303XX	
424		Italic	Exact	C04304XX	
424		Italic	Exact	C04305XX	
424*		Italic	Exact	C04307XX	
425	96	Double Wide Italic	Exact	C0S0CW15	
428		Bold Italic	Exact	C04500XX	C0S0CIXX
428	115	Bold Italic	Exact	C0450000	C0S0CI15
428		Bold Italic	Exact	C04502XX	
428		Bold Italic	Exact	C04503XX	
428		Bold Italic	Exact	C04504XX	
428		Bold Italic	Exact	C04505XX	
428*		Bold Italic	Exact	C04507XX	
432*		Normal	Exact	C07200XX	C0S0CRXX
434	177	Bold	Exact	C0E0OB08	C0S0CB10
435	221	Bold	Exact	C0E0OB06	C0S0CB10
751	53	Normal	Exact	C0T05580	C0D0GT24
751	53	Normal	Exact	C0S0SYM0	
752	80	Normal	Exact	C0E20NB0	C0N204B0
753	120	Bold	Exact	C0E0BNTR	C0N404B0
753	80	Bold	Exact	C0E40NB0	C0N404B0
754	120	Bold	Exact	C0E40NH0	C0N404H0
755	160	Bold	Exact	C0E40NN0	C0N404N0
756	80	Italic	Exact	C0E30NB0	C0N304B0
757	80	Bold Italic	Exact	C0E50NB0	C0N504B0
758	120	Bold Italic	Exact	C0E50NH0	C0N504H0
759	160	Bold Italic	Exact	C0E50NN0	C0N504N0

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
1051	67	Normal	Exact	C0T05500	C0D0GT20
1051	67	Normal	Exact	C0S0SYM0	
1053	67	Bold	Exact	C0T07500	C0D0GT20
1053	67	Bold	Exact	C0S0SYM0	
1056	67	Italic	Exact	C0T15500	C0D0GT20
1056	67	Italic	Exact	C0S0SYM0	
1351	80	Normal	Exact	C0T055B0	C0D0GT18
1351	80	Normal	Exact	C0S0SYM2	
1653	107	Bold	Exact	C0T075F0	C0S0CB15
1653	107	Bold	Exact	C0S0SYM2	
1803	120	Bold	Exact	C0T075H0	C0S0CB10
2103	160	Bold	Exact	C0T075N0	C0S0CB10
2304		Normal	Exact	C0H200XX	C050CRXX
2304	96	Normal	Exact	C0H20080	C0D0GT24
2304	115	Normal	Exact	C0H200H0	C0S0CR12
2304	144	Normal	Exact	C0H200B0	C0D0GT18
2304	169	Normal	Exact	C0H200D0	C0S0CR15
2304	221	Normal	Exact	C0H200J0	C0S0CR10
2304	288	Normal	Exact	C0H200Z0	C0S0CR10
2304		Normal	Exact	C0H202XX	
2304		Normal	Exact	C0H203XX	
2304		Normal	Exact	C0H204XX	
2304		Normal	Exact	C0H205XX	
2304*		Normal	Exact	C0H201XX	
2304*		Normal	Exact	C0H207XX	
2305		Bold	Exact	C0H400XX	C0S0CBXX
2305	96	Bold	Exact	C0H40080	C0D0GT24
2305	115	Bold	Exact	C0H400H0	C0S0CB12
2305	144	Bold	Exact	C0H400B0	C0D0GT18
2305	169	Bold	Exact	C0H400D0	C0S0CB15
2305	221	Bold	Exact	C0H400J0	C0S0CB10
2305	288	Bold	Exact	C0H400Z0	C0S0CB10
2305		Bold	Exact	C0H402XX	
2305		Bold	Exact	C0H403XX	
2305		Bold	Exact	C0H404XX	
2305		Bold	Exact	C0H405XX	
2305*		Bold	Exact	C0H401XX	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
2305*		Bold	Exact	C0H407XX	
2306		Italic	Exact	C0H300XX	C0S0CIXX
2306	96	Italic	Exact	C0H30080	C0D0GT24
2306	115	Italic	Exact	C0H300H0	C0S0CI12
2306	144	Italic	Exact	C0H300B0	C0D0GT18
2306	169	Italic	Exact	C0H300D0	C0S0CI15
2306	221	Italic	Exact	C0H300J0	C0S0CI10
2306	288	Italic	Exact	C0H300Z0	C0S0CI10
2306		Italic	Exact	C0H302XX	
2306		Italic	Exact	C0H303XX	
2306		Italic	Exact	C0H304XX	
2306		Italic	Exact	C0H305XX	
2306*		Italic	Exact	C0H307XX	
2307		Bold Italic	Exact	C0H500XX	C050CIXX
2307	96	Bold Italic	Exact	C0H50080	C0D0GT24
2307	115	Bold Italic	Exact	C0H500H0	C0S0CI12
2307	144	Bold Italic	Exact	C0H500B0	C0D0GT18
2307	169	Bold Italic	Exact	C0H500D0	C0S0CI15
2307	221	Bold Italic	Exact	C0H500J0	C0S0CI10
2307	288	Bold Italic	Exact	C0H500Z0	C0S0CI10
2307		Bold Italic	Exact	C0H502XX	
2307		Bold Italic	Exact	C0H503XX	
2307		Bold Italic	Exact	C0H504XX	
2307		Bold Italic	Exact	C0H505XX	
2307*		Bold Italic	Exact	C0H507XX	
2308		Normal	Exact	C0N200XX	C0S0CRXX
2308	96	Normal	Exact	C0N20080	C0D0GT24
2308	115	Normal	Exact	C0N200H0	C0S0CR12
2308	144	Normal	Exact	C0N200B0	C0D0GT18
2308	169	Normal	Exact	C0N200D0	C0S0CR15
2308	221	Normal	Exact	C0N200J0	C0S0CR10
2308	288	Normal	Exact	C0N200Z0	C0S0CR10
2308		Normal	Exact	C0N202XX	
2308		Normal	Exact	C0N203XX	
2308		Normal	Exact	C0N204XX	
2308		Normal	Exact	C0N205XX	
2308*		Normal	Exact	C0N201XX	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
2308*		Normal	Exact	C0N207XX	
2309		Bold	Exact	C0N400XX	C0S0CBXX
2309	96	Bold	Exact	C0N40080	C0D0GT24
2309	115	Bold	Exact	C0N400H0	C0S0CB12
2309	144	Bold	Exact	C0N400B0	C0D0GT18
2309	169	Bold	Exact	C0N400D0	C0S0CB15
2309	221	Bold	Exact	C0N400J0	C0S0CB10
2309	288	Bold	Exact	C0N400Z0	C0S0CB10
2309		Bold	Exact	C0N402XX	
2309		Bold	Exact	C0N403XX	
2309		Bold	Exact	C0N404XX	
2309		Bold	Exact	C0N405XX	
2309*		Bold	Exact	C0N401XX	
2309*		Bold	Exact	C0N407XX	
2310		Italic	Exact	C0N300XX	C0S0CIXX
2310	96	Italic	Exact	C0N30080	C0D0GT24
2310	115	Italic	Exact	C0N300H0	C0S0CI12
2310	144	Italic	Exact	C0N300B0	C0D0GT18
2310	169	Italic	Exact	C0N300D0	C0S0CI15
2310	221	Italic	Exact	C0N300J0	C0S0CI10
2310	288	Italic	Exact	C0N300Z0	C0S0CI10
2310		Italic	Exact	C0N302XX	
2310		Italic	Exact	C0N303XX	
2310		Italic	Exact	C0N304XX	
2310		Italic	Exact	C0N305XX	
2310*		Italic	Exact	C0N307XX	
2311		Bold Italic	Exact	C0N500XX	C0S0CIXX
2311	96	Bold Italic	Exact	C0N50080	C0D0GT24
2311	115	Bold Italic	Exact	C0N500H0	C0S0CI12
2311	144	Bold Italic	Exact	C0N500B0	C0D0GT18
2311	169	Bold Italic	Exact	C0N500D0	C0S0CI15
2311	221	Bold Italic	Exact	C0N500J0	C0S0CI10
2311	288	Bold Italic	Exact	C0N500Z0	C0S0CI10
2311		Bold Italic	Exact	C0N502XX	
2311		Bold Italic	Exact	C0N503XX	
2311		Bold Italic	Exact	C0N504XX	
2311		Bold Italic	Exact	C0N505XX	

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
2311		Bold Italic	Exact	C0N507XX	
4407		Normal	Exact	C0T055XX	C0S0CRXX
4407	42	Normal	Exact	C0T05560	C0D0GT24
4407	54	Normal	Exact	C0T05580	C0D0GT24
4407	66	Normal	Exact	C0T05500	C0D0GT20
4407	72	Normal	Exact	C0T055A0	C0D0GT20
4407	78	Normal	Exact	C0T055B0	C0D0GT18
4427		Bold	Exact	C0T075XX	C0S0CBXX
4427	66	Bold	Exact	C0T07500	C0D0GT20
4427	96	Bold	Exact	C0T075D0	C0S0CB15
4427	108	Bold	Exact	C0T075F0	C0S0CB15
4427	132	Bold	Exact	C0T075J0	C0S0CB10
4427	162	Bold	Exact	C0T075N0	C0S0CB10
4535		Italic	Exact	C0T155XX	C0S0CIXX
4535	66	Italic	Exact	C0T15500	C0D0GT20
4535	72	Italic	Exact	C0T155A0	C0D0GT20
4555		Bold Italic	Exact	C0T175XX	C0S0CIXX
4555	66	Bold Italic	Exact	C0T17500	C0D0GT20
4555	78	Bold Italic	Exact	C0T175B0	C0D0GT18
4555	132	Bold Italic	Exact	C0T175J0	C0S0CI10
4919	40	Normal	Exact	C0E20G60	C0D0GT18
4919	53	Normal	Exact	C0E20G80	C0S0CR15
4919	67	Normal	Exact	C0E20G00	C0S0CR12
4919	80	Normal	Exact	C0E20GB0	C0S0CR10
4939	67	Bold	Exact	C0E40G00	C0S0CB12
4939	93	Bold	Exact	C0E40GD0	C0S0CB10
4939	120	Bold	Exact	C0E40GH0	C0S0CB10
5047	67	Italic	Exact	C0E30G00	C0S0CI12
5067	67	Bold Italic	Exact	C0E50G00	C0S0CI12
5687	80	Normal	Exact	C0E20TB0	C0S0CR10
5687	67	Normal	Exact	C0E20T00	C0S0CR12
5687	53	Normal	Exact	C0E20T80	C0S0CR15
5687	40	Normal	Exact	C0E20T60	C0D0GT18
5707	160	Bold	Exact	C0E40TN0	C0S0CB10
5707	120	Bold	Exact	C0E40TH0	C0S0CB10
5707	93	Bold	Exact	C0E40TD0	C0S0CB10
5707	80	Bold	Exact	C0E40TB0	C0S0CB10

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
5707	67	Bold	Exact	C0E40T00	C0S0CB12
5815	80	Italic	Exact	C0E30TB0	C0S0CI10
5815	67	Italic	Exact	C0E30T00	C0S0CI12
5835	80	Bold Italic	Exact	C0E50TB0	C0S0CI10
5835	67	Bold Italic	Exact	C0E50T00	C0S0CI12
5943	120	Normal	Exact	C0E20MH0	C0S0CR10
5943	93	Normal	Exact	C0E20MD0	C0S0CR10
5943	80	Normal	Exact	C0E20MB0	C0S0CR10
6199	80	Normal	Exact	C0E20PB0	C0S0CR10
6199	67	Normal	Exact	C0E20P00	C0S0CR12
6199	53	Normal	Exact	C0E20P80	C0S0CR15
6199	40	Normal	Exact	C0E20P60	C0D0GT18
6219	120	Bold	Exact	C0E40PH0	C0S0CB10
6219	93	Bold	Exact	C0E40PD0	C0S0CB12
6219	67	Bold	Exact	C0E40P00	C0S0CB15
6327	67	Italic	Exact	C0E30P00	C0S0CI12
6347	67	Bold Italic	Exact	C0E50P00	C0S0CI12
8503	80	Normal	Exact	C0E20BB0	C0S0CR10
8503	67	Normal	Exact	C0E20B00	C0S0CR10
8503	53	Normal	Exact	C0E20B80	C0S0CR15
8503	40	Normal	Exact	C0E20B60	C0D0GT18
8523	120	Bold	Exact	C0E40BH0	C0S0CB10
8523	93	Bold	Exact	C0E40BD0	C0S0CB10
8523	67	Bold	Exact	C0E40B00	C0S0CB12
8631	67	Italic	Exact	C0E30B00	C0S0CI12
8651	67	Bold Italic	Exact	C0E50B00	C0S0CI12
12855	80	Normal	Exact	C0E20KB0	C0S0CR10
12855	67	Normal	Exact	C0E20K00	C0S0CR12
12855	53	Normal	Exact	C0E20K80	C0S0CR15
12875	160	Bold	Exact	C0E40KN0	C0S0CB10
12875	120	Bold	Exact	C0E40KH0	C0S0CB10
12875	67	Bold	Exact	C0E40K00	C0S0CB12
12875	53	Bold	Exact	C0E40K80	C0S0CB15
12875	80	Bold	Exact	C0E40KB0	C0S0CB10
16951	80	Normal	Exact	C0E20CB0	C0S0CR10
16951	67	Normal	Exact	C0E20C00	C0S0CR10
16951	53	Normal	Exact	C0E20C80	C0S0CR15

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
16951	40	Normal	Exact	C0E20C60	C0D0GT18
16971	120	Bold	Exact	C0E40CH0	C0S0CB10
16971	93	Bold	Exact	C0E40CD0	C0S0CB10
16971	67	Bold	Exact	C0E40C00	C0S0CB12
17079	67	Italic	Exact	C0E30C00	C0S0CI12
17099	67	Bold Italic	Exact	C0E50C00	C0S0CI12
33079		Normal	Exact	C0A055XX	C0S0CRXX
33099		Bold	Exact	C0A075XX	C0S0CBXX
33207		Italic	Exact	C0A155XX	C0S0CIXX
33227		Bold Italic	Exact	C0A175XX	C0S0CIXX
33335	80	Normal	Exact	C0E20OB0	C0S0CR10
33335	67	Normal	Exact	C0E20O00	C0S0CR12
33335	53	Normal	Exact	C0E20O80	C0S0CR15
33335	40	Normal	Exact	C0E20O60	C0D0GT18
33355	120	Bold	Exact	C0E40OH0	C0S0CB10
33355	93	Bold	Exact	C0E40OD0	C0S0CB10
33355	67	Bold	Exact	C0E40O00	C0S0CB12
33463	67	Italic	Exact	C0E30O00	C0S0CI10
33483	67	Bold Italic	Exact	C0E50O00	C0S0CI12
33591	80	Normal	Exact	C0E20FB0	C0S0CR10
33591	67	Normal	Exact	C0E20F00	C0S0CR12
33591	53	Normal	Exact	C0E20F80	C0S0CR15
33591	40	Normal	Exact	C0E20F60	C0D0GT18
33601	120	Bold	Exact	C0E40FH0	C0S0CB10
33601	93	Bold	Exact	C0E40FD0	C0S0CB10
33601	67	Bold	Exact	C0E40F00	C0S0CB12
33719	67	Italic	Exact	C0E30F00	C0S0CI12
33729	67	Bold Italic	Exact	C0E50F00	C0S0CI12
34103	80	Normal	Exact	C0E20HB0	C0S0CR10
34103	67	Normal	Exact	C0E20H00	C0S0CR10
34103	53	Normal	Exact	C0E20H80	C0S0CR15
34103	40	Normal	Exact	C0E20H60	C0D0GT18
34123	120	Bold	Exact	C0E40HH0	C0S0CB10
34123	93	Bold	Exact	C0E40HD0	C0S0CB10
34123	67	Bold	Exact	C0E40H00	C0S0CB12
34231	67	Italic	Exact	C0E30H00	C0S0CI12
34251	67	Bold Italic	Exact	C0E50H00	C0S0CI10

Table 40. Printer Resident to Host Resident Font Character Set Mapping (continued)

Registered Font ID	Font Width	Font Attributes	Map Fidelity	Font Character Set Name (first choice)	Font Character Set Name (second choice)
37431	120	Normal	Exact	C0E20EH0	C0S0CR10
37431	93	Normal	Exact	C0E20ED0	C0S0CR10
37431	80	Normal	Exact	C0E20EB0	C0S0CR10
41783	80	Italic	Exact	C0E30SB0	C0S0CI10
41803	120	Bold Italic	Exact	C0E50SH0	C0S0CI10
41803	93	Bold Italic	Exact	C0E50SD0	C0S0CI10
49719	54	Normal	Exact	C0P05580	C0D0GT18

Printer Resident to Host Resident Code Page Mapping

The following table can help you determine what host resident code page will be downloaded to a 3820, 3825, 3827, 3829, 3831, 3835, or 3900 printer when your spooled file references a registered code page identifier (ID) instead of a host resident code page.

This font substitution is necessary because these printers do not support printer resident fonts. Depending upon the registered code page ID value that is requested for a particular font reference, the appropriate host resident code page is selected to match (as closely as possible) your font request.

The first choice is used if it is present on your iSeries server. The second choice is used if the first choice cannot be found.

The Map Fidelity indicates whether or not the first choice is considered to be an exact match to the printer resident font that is requested in your spooled file. As a rule, the second choice is not considered to be an exact match.

Table 41. Printer Resident to Host Resident Code Page Mapping

Registered Code Page ID	Host Resident Code Page Name (first choice)	Host Resident Code Page Name (second choice)	Map Fidelity
29	T1V10871		Exact
37	T1V10037		Exact
38	T1V10500		Exact
256	T1GDP256		Exact
259	T1000259		Exact
260	T1V10037		Exact
273	T1V10273		Exact
274	T1V10274		Exact
275	T1V10275		Exact
277	T1V10277		Exact
278	T1V10278		Exact
280	T1V10280		Exact

Table 41. Printer Resident to Host Resident Code Page Mapping (continued)

Registered Code Page ID	Host Resident Code Page Name (first choice)	Host Resident Code Page Name (second choice)	Map Fidelity
281	T1V10281		Exact
282	T1V10282		Exact
283	T1V10284		Exact
284	T1V10284		Exact
285	T1V10285		Exact
286	T1V10273		Exact
287	T1V10277		Exact
288	T1V10278		Exact
289	T1V10284		Exact
290	T1V10290		Exact
293	T1000293	T1S0AE10	Exact
297	T1V10297		Exact
310	T1000310	T1S0AE10	Exact
340	T1L0OCR1	T1V10500	Not Exact
361	T1000361	T1GI0361	Exact
363	T1GPI363		Exact
382	T1000382	T1GI0382	Exact
383	T1000383	T1GI0383	Exact
384	T1000384	T1GI0384	Exact
385	T1000385	T1GI0385	Exact
386	T1000386	T1GI0386	Exact
387	T1000387	T1GI0387	Exact
388	T1000388	T1GI0388	Exact
389	T1000389	T1GI0389	Exact
390	T1000390	T1GI0390	Exact
391	T1000391	T1GI0391	Exact
392	T1000392	T1GI0392	Exact
393	T1000393	T1GI0393	Exact
394	T1000394	T1GI0394	Exact
395	T1000395	T1GI0395	Exact
396	T1GI0396		Exact
420	T1000420	T1V10500	Not Exact
423	T1000423		Exact
424	T1000424	T1V10500	Not Exact
437	T1000437	T1V10500	Not Exact
500	T1V10500		Exact
803	T1000803		Exact
813	T1000813		Exact

Table 41. Printer Resident to Host Resident Code Page Mapping (continued)

Registered Code Page ID	Host Resident Code Page Name (first choice)	Host Resident Code Page Name (second choice)	Map Fidelity
819	T1000819		Exact
829	T1M00829		Exact
831	T1V10282		Exact
838	T1000838		Exact
850	T1000850		Exact
851	T1000851		Exact
852	T1000852		Exact
853	T1000853		Exact
855	T1000855		Exact
856	T1000856		Exact
857	T1000857		Exact
860	T1000860		Exact
861	T1000861		Exact
862	T1000862		Exact
863	T1000863		Exact
864	T1000864		Exact
865	T1000865		Exact
866	T1000866		Exact
869	T1000869		Exact
870	T1000870	T1V10500	Not Exact
871	T1V10871		Exact
874	T1V10874		Exact
875	T1000875		Exact
880	T1000880		Exact
890	T1000890	T1V10500	Not Exact
892	T1L0OCR1	T1V10500	Not Exact
893	T1L0OCRB	T1V10500	Not Exact
897	T1000897		Exact
899	T1000899		Exact
905	T1000905		Exact
912	T1000912		Exact
914	T1000914		Exact
915	T1000915		Exact
916	T1000916		Exact
920	T1000920		Exact
1002	T1001002	T1D0BASE	Exact
1003	T1DCDCFS		Exact
1004	T1001004		Exact

Table 41. Printer Resident to Host Resident Code Page Mapping (continued)

Registered Code Page ID	Host Resident Code Page Name (first choice)	Host Resident Code Page Name (second choice)	Map Fidelity
1008	T1001008		Exact
1025	T1001025		Exact
1026	T1001026		Exact
1027	T1001027		Exact
1028	T1001028		Exact
1029	T1001029		Exact
1038	T1001038		Exact
1039	T1001039		Exact
1041	T1001041		Exact
1046	T1001046		Exact
1068	T1001068		Exact
1069	T1001069		Exact
1070	T1GDP037		Exact
1071	T1GDP273		Exact
1072	T1GDP274		Exact
1073	T1GDP275		Exact
1074	T1GDP277		Exact
1075	T1GDP278		Exact
1076	T1GDP280		Exact
1077	T1GDP281		Exact
1078	T1GDP282		Exact
1079	T1GDP284		Exact
1080	T1GDP285		Exact
1081	T1GDP279		Exact
1087	T1001087		Exact
1091	T1001091		Exact
1092	T1001092		Exact
2063	T1D0BASE		Exact
2064	T1GDP276		Exact
2065	T1GI0361		Exact
2066	T1GPI363		Exact
2067	T1GI0382		Exact
2068	T1GI0383		Exact
2069	T1GI0384		Exact
2070	T1GI0385		Exact
2071	T1GI0386		Exact
2072	T1GI0387		Exact
2073	T1GI0388		Exact

Registered Code Page ID	Host Resident Code Page Name (first choice)	Host Resident Code Page Name (second choice)	Map Fidelity
2074	T1GI0389		Exact
2075	T1GI0390		Exact
2076	T1GI0391		Exact
2077	T1GI0392		Exact
2078	T1GI0394		Exact
2079	T1GI0395		Exact
2081	T1GE0200		Exact
2082	T1GE0300		Exact
2086	T1L0OCRB		Exact
2087	T1L0OCR1		Exact
2092	T1S0S193		Exact
2093	T1S0S198		Exact
2102	T1L02773		Exact
2103	T1L02774		Exact
2108	T1S0AE10		Exact

Character Identifier (CHRID) Values Supported

The following table lists all the character identifiers, the related national language groups, the correct code page, and which printers support which character identifier.

Table 42. CHRID Values and Applicable Printers (CHRID Parameter)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Major Groups										
International (and US ASCII)	103 038	500	Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
Multinational	697 500		Yes	Yes	Yes	Yes	N/A	N/A	Yes	Yes
	337 256	500	Yes	N/A	N/A	N/A	N/A	Yes	Yes	Yes
	697 256	500	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
United States	101 037 697 037		Yes Yes	Yes Yes	Yes N/A	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
US, Canada, Netherlands, Portugal, Brazil, Australia, New Zealand[10]	695 1140	697 037	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Individual Countries/Languages										
Arabic	697 361		Yes	N/A	Yes	N/A	N/A	N/A	Yes	N/A
Arabic X/B	235 420	500	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
	697 420		Yes	N/A	4224-No 4230-Yes 4247-Yes	IPDS[7]	N/A	N/A	Yes	N/A
Arabic [10]	1461 420	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Austria/ Germany[6]	265 273 697 273		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Austria/ Germany	697 286 317 286	273	Yes Yes	N/A N/A	Yes Yes	IPDS[7] IPDS[7]	N/A N/A	N/A N/A	N/A N/A	Yes Yes
Austria, Germany[10]	695 1141	697 273	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Belgium[6]	697 500 269 274 697 274		N/A N/A N/A	Yes Yes Yes	Yes Yes Yes	Yes Yes Yes	Yes Yes N/A	Yes Yes N/A	N/A N/A N/A	Yes Yes Yes
Belgium, Canada, Switzerland[10]	695 1148	697 500	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Brazil[6]	273 275 697 275		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Bulgaria, FYR Macedonia, Serbia (Cyrillic)[10]	1381 1154	1150 1025	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32[5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Canadian French[6]	277 276 341 260	297 037	Yes Yes	N/A N/A	N/A Yes	N/A IPDS[7]	Yes N/A	Yes N/A	Yes Yes	Yes Yes
	697 260		Yes	N/A	N/A	IPDS[7]	N/A	N/A	Yes	N/A
Canada-Bilingual	038 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
	039 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Canada-English	037 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Chinese (Hong Kong S.A.R.)	119 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
Chinese-Simplified	1174 836		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Chinese-Traditional	1175 037		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Chinese-Traditional[10]	32000 1159	697 37	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Cyrillic	960 880		N/A	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
Cyrillic multilingual	1150 1025				4224-No 4230-Yes 4247-Yes	IPDS[7]				Yes
Czechoslovakia/ Czech	083 257		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Czechoslovakia/ Slovak	085 257		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Czech Republic, Hungary, Poland[10]	1375 1153	959 870	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Denmark/ Norway[6]	281 277 697 277		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Denmark/ Norway	697 287	277	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
	321 287		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
Denmark, Norway[10]	695 1142	697 277	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Farsi	1219 1097		N/A	N/A	Yes	IPDS[7]	N/A	N/A	N/A	N/A
Estonia	1307 1122		N/A	N/A	4224-No 4230-Yes 4247-Yes	N/A	N/A	N/A	N/A	N/A
Estonia[10]	1391 1157	1307 1122	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Finland/ Sweden[6]	285 278 697 278		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Finland/ Sweden	697 288	278	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
	325 288		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
Finland, Sweden[10]	695 1143	697 278	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
France (1977)[6]	289 279	297	Yes	N/A	N/A	N/A	N/A	Yes	Yes	N/A
France (1980)[6]	288 297 697 297		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	N/A N/A	Yes Yes	Yes Yes
France	251 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
France/ Belgium	031 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
France[10]	695 1147	697 297	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Germany/ Austria	028 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
	029 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Greek	218 423		N/A	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
	925 875		N/A	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
Greek[10]	1371 875	218 423	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Language Groups	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32[5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Hebrew	941 424		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
	697 424		Yes	N/A	4224-No 4230-Yes 4247-Yes	IPDS[7]	N/A	N/A	Yes	N/A
	1147 803		N/A	N/A	4224-No 4230-Yes 4247-Yes	IPDS[7]	N/A	N/A	N/A	Yes
Hebrew[10]	1356 424	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Hungary	091 257		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Icelandic	697 871		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
	697 029		Yes	N/A	N/A	N/A	N/A	N/A	Yes	N/A
Iceland[10]	695 1149	697 871	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Italy[6]	293 280 697 280		Yes Yes	Yes Yes	Yes Yes	IPDS[7] Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Italy	041 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Italy[10]	695 1144	697 280	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Japan-English[6]	297 281 697 281		Yes Yes	Yes Yes	Yes Yes	IPDS[7] Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
	068 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
	069 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Japan-Katakana[6]	332 290		Yes	N/A	Yes	Yes	N/A	Yes	Yes	Yes
Japan- Katakana	1172 290		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Japan-Katakana[10]	1398 290		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Japan-Latin	1172 1027		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Japan-Latin[10]	1398 1027		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Korean	1173 833		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Korean	933 833		N/A	N/A	4230-Yes 4247-Yes 4224-N/A	IPDS[7]	N/A	N/A	N/A	N/A
	697 290		Yes	N/A	N/A	IPDS[7]	N/A	N/A	Yes	N/A
Latin	959 870		N/A	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
Latin America/ Puerto Rico	025 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Latvia/Lithuania	1305 1112		N/A	N/A	4224-No 4230-Yes 4247-Yes	N/A	N/A	N/A	N/A	N/A
Latvia/Lithuania[10]	1393 1156	1305 1112	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Lao	1341 1132		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Netherlands	043 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Norway/ Denmark	055 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Poland	093 257		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A

| Language Groups | Code Pages | | Printers[1] | | | | | | | |
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Portugal[6]	301 282 697 282		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
Portugal	697 831 063 256	282	Yes Yes	N/A N/A	Yes N/A	N/A N/A	N/A Yes	N/A N/A	Yes Yes	N/A N/A
Romania	087 258		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
South Africa	081 258		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Spain[6]	305 283 697 283	284 284	Yes Yes	N/A N/A	Yes Yes	Yes N/A	Yes N/A	Yes N/A	Yes Yes	Yes Yes
	697 289	284	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
	329 289		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
	045 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Spanish Speaking[6]	309 284 697 284 1149 284		Yes Yes N/A	Yes Yes N/A	Yes Yes N/A	Yes Yes N/A	Yes Yes Yes	Yes N/A N/A	Yes Yes N/A	Yes N/A Yes
Spain, Latin America (Spanish)[10]	695 1145	697 284	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Sweden/ Finland	052 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
	053 256		Yes	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Switzerland/ French	048 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Switzerland/ German	049 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Thai	1102 889		N/A	N/A	Yes	IPDS[7]	N/A	N/A	N/A	N/A
	938 838		N/A	N/A	4230-Yes 4247-Yes 4224-N/A	IPDS[7]	N/A	N/A	N/A	N/A

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32[5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Thai[10]	1395 1160	938 838	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Turkish	965 905		N/A	N/A	4230-Yes 4247-Yes 4224-Yes	IPDS[7]	N/A	N/A	Yes	Yes
	1152 1026		N/A	N/A	4230-Yes 4247-Yes 4224-N/A	IPDS[7]	N/A	N/A	N/A	Yes
Turkish[10]	1378 1155	1152 1026	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Ukraine	1326 1123	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Ukraine[10]	1388 1158	1326 1123	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
United Kingdom[6]	313 285 697 285		Yes Yes	Yes Yes	Yes Yes	Yes Yes	Yes N/A	Yes N/A	Yes Yes	Yes Yes
U.K./ Israel	066 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
U.K./ Israel-Latin	067 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
United Kingdom[10]	695 1146	697 285	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
USA-Accounting	017 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
USA/Australia	001 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Vietnamese	1336 1130		N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
Vietnamese[10]	1397 1164	1336 1130	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32[5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Countries of the former Yugoslavia	410 890		N/A	N/A	Yes	IPDS[7]	N/A	N/A	N/A	N/A
Countries of the former Yugoslavia-Latin	095 257		N/A	N/A	N/A	N/A	Yes	N/A	N/A	N/A
Noncountry Languages										
APL	697 293		Yes	N/A	N/A	IPDS[7]	N/A	N/A	Yes	N/A
	380 293		Yes	N/A	4224-N/A 4230-N/A 4247-N/A	IPDS[7]	N/A	N/A	Yes	Yes
APL Alternate	697 310		Yes	N/A	4224-No 4230-Yes 4247-Yes	IPDS[7]	N/A	N/A	Yes	Yes
	963 310		Yes	N/A	4224-No 4230-Yes 4247-Yes	N/A	N/A	N/A	Yes	Yes
ASCII	103 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
DCF Compatibility	1132 1002		Yes	N/A	4230-Yes 4247-Yes 4224-No	IPDS[7]	N/A	N/A	Yes	Yes
DCF US Text	1133 1003		N/A	N/A	4230-N/A 4247-N/A 4224-N/A	N/A	N/A	N/A	N/A	Yes

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
DCF text with numeric space	1259 1068		N/A	N/A	4230-N/A 4247-N/A 4224-N/A	N/A	N/A	N/A	N/A	Yes
EBCDIC	101 256		Yes	N/A	N/A	N/A	Yes	N/A	Yes	Yes
GML List Symbols	1258 1039		N/A	N/A	N/A	N/A	N/A	N/A	N/A	Yes
International Typographic	697 361		Yes	N/A	N/A	N/A	N/A	N/A	Yes	N/A
OCR (unregistered)	697 340	500	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
OCR A	697 892	500	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
	968 892		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
OCR A (unregistered)	580 340	892	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
OCR B	697 893	500	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
	969 893		Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	Yes
OCR B (unregistered)	590 340	893	Yes	N/A	Yes	IPDS[7]	N/A	N/A	Yes	N/A
Personal Computer	697 437		Yes	N/A	4224-No 4247-Yes 4230-Yes	N/A	N/A	N/A	Yes	N/A
Symbols	340 259		Yes	N/A	N/A	N/A	N/A	N/A	Yes	Yes
Symbol-Selectric	201 259	500	Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Symbol-6640	202 259	500	Yes	Yes	N/A	N/A	Yes	N/A	Yes	N/A
Symbol-6670	203 259		Yes	N/A	N/A	N/A	Yes	N/A	Yes	N/A
Symbols, Adobe	1257 1087		N/A	N/A	N/A	N/A	N/A	N/A	N/A	Yes

Table 42. CHRID Values and Applicable Printers (CHRID Parameter) (continued)

Language Groups	Code Pages		Printers[1]							
	CHRID Code Page xxx yyy[2,3]	Sub-stitute Code Page yyy[2,4]	3812[5] 3816[5]	4214[5]	4224[5] 4230[5] 4247[5]	4234[5] 6400[9] 6408[9] 6412[9]	5219	5224 5225	3112 3116 3912 3916 4312 4317 4324 4028 InfoPrint 20 InfoPrint 32 [5]	3130 3160 3935 InfoPrint 3000 InfoPrint 4000
Symbols Set 7	697 259		Yes	N/A	N/A	N/A	N/A	N/A	Yes	N/A
Symbols Mod Set 7	1191 1091		N/A	N/A	N/A	N/A	N/A	N/A	N/A	Yes
Symbols Set 8	630 363		N/A	N/A	N/A	N/A	N/A	N/A	N/A	Yes

:

[1] The 5256, 5262, and 4245 work station printers do not support the hardware function required for alternative CHRID processing. If a nondefault character set and code page is selected for these printers, a diagnostic message is sent and processing continues using the default character set.

[2] If the printer supports the code page specified (the second part *(yyy)* of the CHRID parameter) but not the character set *(xxx)*, then the character set supported by the printer is used along with the specified code page. For example, if 337 037 (extended character set for displays) is specified for the 5224 and 5225 Printers, the print file is printed with character set 101, code page 037.

[3] In some cases, the printer will substitute a supported code page for an unsupported code page. Consult the various printer reference guides for defaults on the code page mapping.

[4] If the printer does not support or map the code page specified, an attempt is made by the system to find a satisfactory substitute. This column shows the code page substitutes that are made if the specified printer supports the substitute.

[5] The 3812, 3816, 4214, 4224, 4230, 4234, and 4247 Printers support character set 697 (full character set). This character set contains all the characters in the limited character sets. For example, 697 037 would contain all the characters in 101 037 or 337 037 (extended character set for displays).

[6] This language is considered a primary language group. All other entries, if any, under the primary language group are considered as alternative language groups.

[7] Supported by 4234 IPDS version and 64xx with IPDS feature only.

[8] Supported by 4234 SCS version only.

[9] 64xx SCS emulation mode set to 4234

[10] Provides support for Euro currency symbol.

Host Resident to Printer Resident Code Page Mapping

Code pages are necessary for jobs to print. Some printers have code pages stored in memory or on a font card (printer resident code page); other printers do not.

Refer to the following table:
- if your spooled files are directed to a 4224, 4230, 4234, 4247, or 64xx printer that is configured as an AFP printer.
- if your spooled files have a host resident code page (stored on the iSeries server) specified.

The table can help you determine what printer resident code page is substituted when your spooled file specifies a host resident code page instead of a registered code page identifier (ID).

This code page substitution is necessary because these printers do not support the downloading of 240-pel host resident fonts. Depending upon the host resident code page name that is requested for a particular font reference, the appropriate registered code page ID value is selected to match (as closely as possible) your font request.

The Map Fidelity indicates whether or not the substituted printer resident code page is considered to be an exact match to the code page that is requested in your spooled file.

Table 43. Host Resident to Printer Resident Code Page Mapping

Host Resident Code Page Name	Registered Code Page ID	Map Fidelity
T1V10037	37	Exact
T1GDP256	256	Exact
T1V10273	273	Exact
T1V10274	274	Exact
T1V10275	275	Exact
T1V10277	277	Exact
T1V10278	278	Exact
T1V10280	280	Exact
T1V10281	281	Exact
T1V10282	282	Exact
T1V10284	284	Exact
T1V10285	285	Exact
T1000290	290	Exact
T1L02773	2102	Exact
T1L02774	2103	Exact
T1S0AE10	2108	Exact
T1V10297	297	Exact
T1000361	361	Exact
T1GI0361	2065	Exact
T1000382	382	Exact
T1GI0382	2067	Exact
T1GI0383	2068	Exact
T1000384	384	Exact
T1GI0384	2069	Exact
T1000385	385	Exact
T1GI0385	2070	Exact
T1000386	386	Exact
T1GI0386	2071	Exact
T1000387	387	Exact
T1GI0387	2072	Exact
T1000388	388	Exact
T1GI0388	2073	Exact

Host Resident Code Page Name	Registered Code Page ID	Map Fidelity
T1000389	389	Exact
T1GI0389	2074	Exact
T1000390	390	Exact
T1GI0390	2075	Exact
T1000391	391	Exact
T1GI0391	2076	Exact
T1000392	392	Exact
T1GI0392	2077	Exact
T1000393	393	Exact
T1GI0393	2077	Exact
T1000394	394	Exact
T1GI0394	2078	Exact
T1000395	395	Exact
T1GI0395	2079	Exact
T1000420	420	Exact
T1000424	424	Exact
T1GPI363	2066	Exact
T1000437	437	Exact
T1V10500	500	Exact
T1000819	819	Exact
T1000850	850	Exact
T1000852	852	Exact
T1000857	857	Exact
T1000863	863	Exact
T1000870	870	Exact
T1V10871	871	Exact
T1000912	912	Exact
T1000920	920	Exact
T1001002	1002	Exact
T1D0BASE	1002	Exact
T1001003	1003	Exact
T1DCDCFS	1003	Exact
T1001004	1004	Exact
T1001026	1026	Exact

Lines Per Inch (LPI) Values Supported

Lines per inch means the number of characters that can be printed vertically within an inch.

Each entry in the following table shows the valid range of values for lines per page for each printer type and for each value of lines per inch (LPI) valid for the printer.

Note: Because of slight adjustments made for position checks, it is recommended to not print on line 1 when specifying 8 or 9 LPI on an IPDS printer.

Table 44. Lines per Page (LPI Parameter)

Printer	3 Lines per Inch	4 Lines per Inch	6 Lines per Inch	7.5 Lines per Inch	8 Lines per Inch	9 Lines per Inch	12 Lines per Inch
3287	–	1-104	1-104	–	1-104	–	–
3812 SCS	–	1-56	1-84	–	1-112	1-126	1-168
3812 IPDS	–	2-56	2-84	–	2-112	2-112	2-168
3816 SCS	–	1-56	1-84	–	1-112	1-126	1-168
3816 IPDS	–	2-56	2-84	–	2-112	2-112	2-168
3820	–	1-56	1-84	–	1-112	1-126	1-168
3825	–	1-56	1-84	–	1-112	1-126	1-168
3827	–	1-56	1-84	–	1-112	1-126	1-168
3835	–	2-91	2-136	–	2-182	2-204	2-273
3935		1-68	1-102		1-136	1-153	1-204
4028	–	2-56	2-84	–	1-112	1-112 or 2-126	2-168
4214	–	1-255	1-255	–	1-255	1-255	–
4224, 4234 IPDS	–	2-91	2-136	–	2-182	2-204	2-273
4230	–	2-91	2-136	–	2-182	2-204	2-273
4234 SCS	–	1-255	1-255	–	1-255	–	–
4245 Models T12 and T20	–	–	1-255	–	1-255	–	–
4247	–	2-91	2-136	–	2-182	2-204	2-273
5211	–	–	2-84	–	2-112	–	–
5219 Continuous Forms	–	2-255	2-255	–	2-255	–	2-255
5219 Cut Sheet	–	57	86	–	114	–	172
5224	–	1-255	1-255	–	1-255	1-255	–
5225	–	1-255	1-255	–	1-255	1-255	–
5256 (set manually)	–	–	1-255	–	1-255	–	–
5262	–	–	1-255	–	1-255	–	–
5553	1-255	1-255	1-255	1-255	1-255	–	1-255
5583	1-255	1-255	1-255	1-255	1-255	–	–
6252	–	1-255	1-255	–	1-255	1-255	–

Characters Per Inch (CPI) Values Supported

Characters per inch means the number of characters printed horizontally within an inch across a page.

Each entry in the following table shows the valid range of values for the characters per line for each printer type and for each value of characters per inch (CPI) for the printer.

Table 45. Characters per Line (CPI Parameter)

Printer	5 Characters per Inch	10 Characters per Inch	12 Characters per Inch	13.3 Characters per Inch	15 Characters per Inch	16.7 Characters per Inch	18 Characters per Inch	20 Characters per Inch
3287	–	1-132	–	–	–	–	–	–
3112[1]	1-42	1-85	1-102	–	1-127	–	–	–
3116[1]	1-42	1-85	1-102	–	1-127	–	–	–
3130[1]	–	1-132	1-158	–	1-198	–	–	–
3160[1]	–	1-132	1-158	–	1-198	–	–	–
3812[1]	1-42	1-85	1-102	–	1-127	–	–	–
3812[1] Rotated Form	1-70	1-140	1-168	–	1-210	–	–	–
3816[1]	1-42	1-85	1-102	–	1-127	–	–	–
3816[1] Rotated Form	1-70	1-140	1-168	–	1-210	–	–	–
3820[1]	–	1-85	1-102	–	1-127	–	–	–
3825[1]	–	1-85	1-102	–	1-127	–	–	–
3827[1]	–	1-85	1-102	–	1-127	–	–	–
3835[1], 3935[1]	–	1-132	1-158	–	1-198	–	–	–
3912[1]	1-42	1-85	1-102	–	1-127	–	–	–
3916[1]	1-42	1-85	1-102	–	1-127	–	–	–
4028[1]	1-42	1-85	1-102	–	1-127	–	–	–
4028[1] Rotated Form	1-70	1-140	1-168	–	1-210	–	–	–
4214 Continuous Forms	1-66	1-132	1-158	–	1-198	1-220	–	–
4214 Cut Sheet	1-60	1-120	1-144	–	1-180	1-200	–	–
4224[1]	–	1-132	1-158	–	1-198	1-220	–	–
4230[1]	–	1-132	1-158	–	1-198	1-220	–	–
4234 SCS[1]	–	1-132	–	–	1-198	–	–	–
4234 IPDS[1]	1-66	1-132	1-158	–	1-198	1-238	–	–
4245	–	1-132	–	–	–	–	–	–
4247[1]	–	1-132	1-158	–	1-198	1-220	–	–
5219	–	1-132	1-158	–	1-198	–	–	–
5224	–	1-132	–	–	1-198	–	–	–
5225	–	1-132	–	–	1-198	–	–	–

Table 45. Characters per Line (CPI Parameter) (continued)

Printer	5 Characters per Inch	10 Characters per Inch	12 Characters per Inch	13.3 Characters per Inch	15 Characters per Inch	16.7 Characters per Inch	18 Characters per Inch	20 Characters per Inch
5256 Model 3	–	1-132	–	–	–	–	–	–
5262	–	1-132	–	–	–	–	–	–
5553	–	1-136	1-163	1-181	1-204	–	1-244	1-272
5583	–	1-132	1-158	1-176	1-198	–	1-236	1-264
6252	–	1-132	–	–	1-198	–	–	–
6408 SCS[2]	–	1-132	–	–	1-198	–	–	–
6408 IPDS[3]	1-66	1-132	1-158	–	1-198	1-238	–	–

:

[1] Many character per inch values (implied by the pitch of the font, see the FONT parameter), are supported in addition to the ones listed here. To find the maximum characters per line, multiply the implied characters per inch value listed in the font table by maximum page width supported (in inches). The maximum page width supported by the 3812 and 3816 Printers is 8.5 inches for non-rotated forms and 14.0 inches for rotated forms.

[2] Emulates 4234 SCS or 5225

[3] Emulates 4234 IPDS

4019 Printer Information

The following tables list the ways you can attach the 4019 printer, the emulation method used, and the function provided by the combination of the attachment and emulation methods. In this table a supported function is indicated by an X.

The 4019 is supported on the iSeries server by treating it as an emulated version of another device. In some respects the result achieved with the 4019 is not identical to the emulated device. The following tables express capabilities in terms of the emulated printers, but indicate some situations in which the 4019 result exceeds that of the emulated device.

Note: Go to Table 30 on page 312 for information on how to work with your emulated 4019 printer to make the 4019 resident fonts available.

Note that image, graphics, and bar codes are not supported under any emulation or means of attachment.

Pay special attention to the treatment of fonts and the footnotes relating to page length and width.

Table 46 shows a matrix of functions when printing via any system function other than OfficeVision. In this table a supported function is indicated by an **X**.

Table 46. 4019 Printer-System Functions

ATTACHED VIA ⇒	3477	3197	AWSC	WSF	WSE	E5250	R5250	OS/2 WSF
EMULATING ⇒	5219	4214	3812	3812	5219	5219	5219	5219
Printer File Commands								
Page Length[1]	X	X	X	X	X	X	X	X

Table 46. 4019 Printer-System Functions (continued)

ATTACHED VIA ⇒ EMULATING ⇒	3477 5219	3197 4214	AWSC 3812	WSF 3812	WSE 5219	E5250 5219	R5250 5219	OS/2 WSF 5219
Page Width[1]	X	X	X	X	X	X	X	X
LPI (4.0)	X	X	X	X	X	X	X	X
LPI (6.0)	X	X	X	X	X	X	X	X
LPI (8.0)	X	X	X	X	X	X	X	X
LPI (9.0)[2]	X	X	X	X				
FONT(*CPI)[3] CPI(5.0)[4]	X	X	X	X				
FONT(*CPI)[3] CPI(10.0)	X	X	X	X	X	X	X	X
FONT(*CPI)[3] CPI(12.0)	X	X	X	X	X	X	X	X
FONT(*CPI)[3] CPI(15.0)	X	X	X	X	X	X	X	X
FONT(*CPI)[3] CPI(16.7)		X	X	X	X	X	X	X
Fold Records	X	X	X	X	X	X	X	X
Truncate Records	X	X	X	X	X	X	X	X
Paper Drawer (1)	X	X	X	X	X	X	X	X
Paper Drawer (2)	X		X	X		X		
Paper Drawer (E1)	X		X	X	X	X		
Non-Typographical Fonts (See Font Table below for details.)								
Typo and User Defined Fonts								
Form Feed (*CUT)		X		X	X		X	X
Form Feed (*AUTOCUT)	X	X	X	X	X	X	X	X
Print Quality (*Draft) with PAGRTT(*DEVD) automatically give PAGRTT (*COR)			X	X				
Change Char Set/Code Page ID	X		X	X				
Rotation 0	X	X	X	X	X	X	X	X
Rotation 90								
Rotation 180								
Rotation 270	X		X	X				
Rotation *COR			X	X				
Print Text	X	X	X	X	X	X	X	X
Hardware Justification 0	X	X	X	X	X	X	X	X

Table 46. 4019 Printer-System Functions (continued)

ATTACHED VIA ⇒ EMULATING ⇒	3477 5219	3197 4214	AWSC 3812	WSF 3812	WSE 5219	E5250 5219	R5250 5219	OS/2 WSF 5219
Hardware Justification 50	X			X	X	X	X	X
Hardware Justification 100	X		X	X	X	X	X	X
Duplex								
Copies	X	X	X	X	X	X	X	X
File Separators	X	X	X	X	X	X	X	X
Additional DDS Keywords								
Barcode								
Chrsiz								
Color								
Font (Changing Type Styles)								
Highlight			X					
Skipa	X	X	X	X	X	X	X	X
Skipb	X	X	X	X	X	X	X	X
Spacea	X	X	X	X	X	X	X	X
Spaceb	X	X	X	X	X	X	X	X
Underline	X	X	X	X	X	X	X	X
Other Functions								
Graphics								
Image								
Symbols Code Page 259	X		X	X				

:

[1] Existing applications or documents may not fit on the 4019 page since there is an unprintable border around the outside edge. You may have to change the margins and lines per page (and re-paginate in some cases) to obtain the desired output.

This unprintable area applies to both envelopes and paper of any size. The unprintable area is 6.35 mm (0.25 in) from the sides and 4.23 mm (0.17 in) from the top and bottom. This results in an 8-inch writing line on 8.5 by 11 inch paper and a 7.7-inch writing line on A4 paper. With 6 LPI for example, this yields 64 lines on an 11-inch page or 68 lines on A4 paper.

Consideration should be given to the effect of this unprintable area when a document is formatted to assure that it prints correctly. If data is formatted to print beyond the printable area on the sides, the excess will print as an additional short line.

[2] LPI(9.0) is not supported by the 5219 printer; therefore LPI(9.0) is not supported for any of the 5219 emulations.

[3] On the CRTPRTF, CHGPRTF, and OVRPRTF commands you may avoid direct specification of a font by using FONT(*CPI). This allows the system to default to any font that supports the requested CPI value. However, the defaulted font may be one that is not supported on the 4019. An unsupported font will cause printing to halt and require operator intervention. We therefore recommend that you specify FONT explicitly on these commands.

[4] CPI(5.0) is not supported by the 5219; therefore CPI(5.0) is not supported for any of the 5219 emulations.

4234 Compressed Font Substitution by Lines Per Inch (LPI) Value

The following table lists the font substitution that takes place when printing on a 4234 printer configured as follows:

Value of *NO for the AFP parameter

Value greater than or equal to 8 for the LPI parameter

This substitution allows the use of fonts that are slightly shorter when the LPI value is greater than or equal to 8.

Table 47. 4234 Compressed Font Substitution by Lines Per Inch (LPI) Value

Font Used When LPI is 4 or 6	Font Substituted When LPI is Greater Than or Equal to 8
11	52
26	51
85	75
87	74
160	154
204	205
222	232
223	233
258	259
400	300

Appendix E. Printer Data Streams

SNA Character String (SCS)

The SNA character string (SCS) has a relatively simple structure, consisting of a 1-byte hexadecimal control code followed by the data to be printed. Multiple control codes can be inserted into SCS by printing applications in order to achieve certain types of output. The SNA character string is sent to the printer in physical blocks of 256 bytes.

The following are a few examples of SCS control codes:

Hex	Description	Code
03	ASCII transparent data	(TRNA)
05	Horizontal tab	(HT)
0B	Vertical tab	(VT)
0C	Forms feed	(FF)
0D	Carriage return	(CR)
1A	Unit backspace	(UBS)
15	New line	(NL)
16	Backspace	(BS)
35	Transparent	(TRN)
2843	Set attribute	(SA)
2BC1	Set horizontal format	(SHF)
2BC2	Set vertical format	(SVF)
2BC6	Set line density	(SLD)
2BD2	Set presentation page size	(SPPS)
2BD4	Begin underscore	(BUS)
2BFE	Load alternate character	(LAC)

How Print Attributes Are Implemented by SCS

The following examples show how common print functions can be implemented by printing applications using SCS.

Line spacing
The equivalent number of NL (new line) control codes is inserted.

Underscoring
The underscored text is first inserted into SCS, followed by the appropriate number of BS (backspace) control codes, and then the same number of underscores.

Overstriking
This is similar to underscoring, except that the overstrike characters are used instead of the underscore.

Emphasis
The text to be emphasized is put into SCS, followed by the necessary number of BS (backspace) control codes and then the emphasized text is repeated. This sequence could be repeated several times.

Page eject
The FF (forms feed) control code is inserted.

Font changes

Font changes are supported by printers such as the 5219 that support final-form text (FFT). Font changes issued by OfficeVision word processing generate a message requesting a print wheel change.

Final-Form Text: Document Content Architecture (FFT DCA)

Final-Form Text: Document Content Architecture is an extension of SCS. It is used within the OfficeVision Office environment and defines how the data streams that represent a document to be printed are organized. Data streams containing final-form text documents must be processed sequentially from beginning to end.
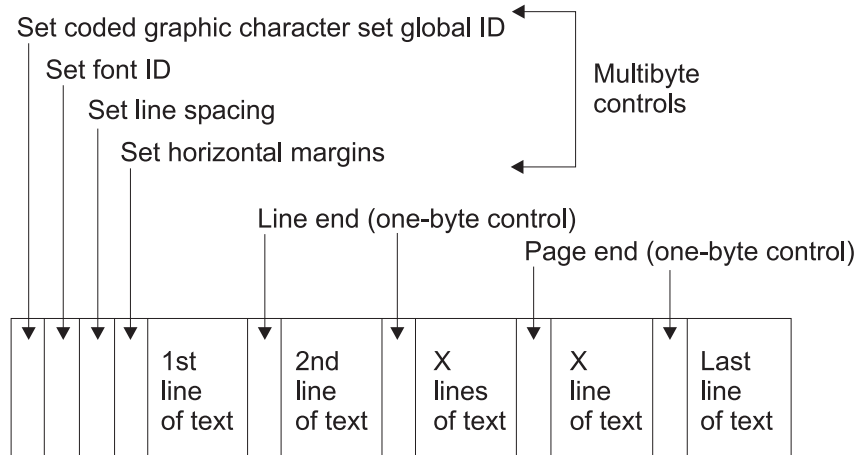
RBAFT529-0

The EBCDIC 1-byte control characters provide functions such as line end, backspace, and indent that have an immediate effect. The multibyte (extended) control characters provide functions such as line spacing, horizontal and vertical margins, and tab settings. These have both a continual and an immediate effect. Some of the multibyte control characters remain in effect only until the line end, but the majority remain in effect until the end of the document or until they are redefined.

When a document is transformed from revisable-form text to final-form text, the formatting declarations specified within the revisable-form text data stream are converted to formatting control codes (1- and multibyte) within the final-form text data stream. These codes are embedded within the text of the document wherever they are needed to format the document. A final-form text data stream can be interpreted by printers that might not contain the function necessary to interpret revisable-form text data streams. These printers produce the same document content in consistent format, subject only to any limitations of the individual printers.

The final-form text document contains formatting control codes at the beginning to establish its initial format. If the control codes are omitted, pre-defined default values are used. These formatting controls are followed by the text of the document, interspersed with the required formatting control codes.

Formatting control codes within a final-form text document control the following functions:
* Top margin location
* Left margin location

- Line spacing
- Font definition
- Justify (align) text
- Begin and end underscore
- Begin and end overstrike.

Advanced Function Printing Data Stream (AFPDS)

AFPDS is the application interface to advanced function printing (AFP) based on the Mixed Object:Document Content Architecture–Presentation (MO:DCA–P). It is independent of both printers and operating systems.

AFPDS is a structured data stream. Structured fields are used to define composed text pages, line format data, and a mixture of line and composed text data. The structured field is a self-identifying string of bytes containing data or parameters and must have an introducer, which contains a length field, an identifier, a flag, and a sequence number. This is followed by parameter bytes that contain control information or data to be printed.
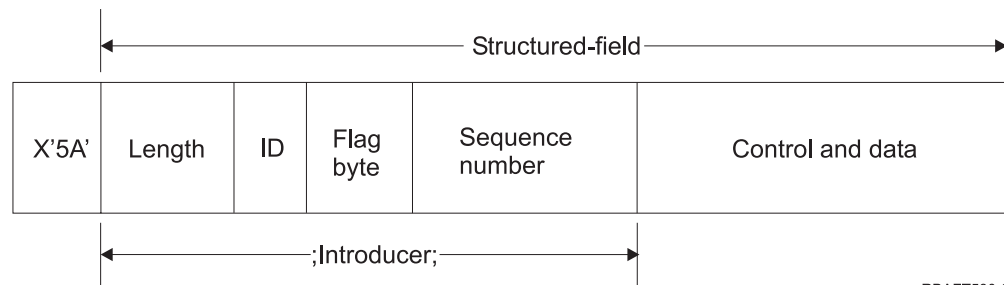
RBAFT530-0

Length
> A 2-byte field that specifies the length of the record (not including the 5A control character).

Identifier (ID)
> A 3-byte field that specifies the type of the structured field.

Flag byte
> A 1-byte field that specifies information about the data field.

Sequence number
> A 2-byte field that identifies the record.

Control and data
> A text control code, the name of an object, or coordinates for positioning an image or page segment. The control information is followed by the data to be printed.

Source Programs That Generate AFPDS

The following IBM licensed programs generate AFPDS data streams:
- Operating System/400 (OS/400)
- Advanced Function Printing Utilities/400 (AFP Utilities/400)
- Document Composition Facility (DCF)
- DisplayWrite/390 (DW/390)
- Graphical Data Display Manager (GDDM)
- System/390 advanced function printing utilities:

- Page Printer Formatting Aid (PPFA)
- Overlay Generation Language (OGL)
- Font Library Service Facility (FLSF)
- Print Management Facility (PMF)
- Print Service Access Facility (PSAF).

Advanced Function Printing

AFPDS describes what the page of data looks like and references printer resource objects by name, assuming that they are resident on the system where the actual printing is to take place and not on the system where the output was spooled. The printer resource objects that are resident on the system are downloaded to the printer by the processor when needed.

The iSeries server uses the following objects when processing AFPDS:

AFPDS spool buffers
> The AFPDS print file is sent to the system in one or more buffers. These buffers contain single or multiple contiguous structured fields. A pointer to the buffer and the length of each buffer are passed to the system.

AFPDS resource objects
> Resource objects contain data and control information that can be used in printing a job. These can be shared by different pages in the same job. A resource is composed entirely of structured fields.
>
> Types of resources are:
> - Fonts
> - Form definitions (FORMDEFs)
> - Page segments
> - Overlays
> - Page definitions
>
> These resources can be transmitted from a host System/390 to an iSeries server or loaded from tape into space objects using OS/400 commands. Overlays and page segments can be created by AFP Utilities/400.

Messages
> Messages generated during the processing of AFPDS print files are placed in the print writer job log.

Intelligent Printer Data Stream (IPDS)

Introduction to IPDS Architecture

Intelligent printer data stream (IPDS) is IBM's Systems Application Architecture host-to-printer data stream for advanced function printing subsystems. It provides an interface to all-points-addressable (APA) printers that makes possible the presentation of pages containing an architecturally unlimited mixture of different data types: high-quality text, raster image, vector graphics, and bar code.

IPDS incorporates the following features:
- Different applications may create source data (graphics, image, bar code, and text), independently of one another. IPDS makes it possible for the output of these independent applications to be merged at print time resulting in an integrated mixed data page.

IPDS makes this possible by carrying independently defined blocks of data *(objects)*. The IBM 3270 display data stream also carries similarly defined independent objects, thus making it possible to use the same objects in both environments.

- IPDS is independent of the carrying communications protocol. This allows the transmission of the same data stream to channel-attached printers, controllers, local area networks, and any other networking link supporting transparent transmission of data.
- IPDS transfers all data and commands through self-identifying structured fields which describe the presentation of the page and provide for the following:
  - Dynamic management of downloaded resources (overlays, page segments, and loaded fonts) and resident fonts
  - Control of device functions such as duplexing, media bin selection, and output finishing
  - Comprehensive handling of exception functions, enabling users to control the level of exception handling.
- IPDS provides an extensive acknowledgement protocol at the data stream level. This acknowledgement protocol helps synchronize host and printer processes, exchange query/reply information, and return detailed exception information.

IPDS Functional Divisions

IPDS architecture is divided into several functional areas, each of which contains a set of IPDS commands representing a major printer capability. This function set design allows IPDS to support a wide range of printer products. Product developers can match function set implementations to the specific needs of their product.
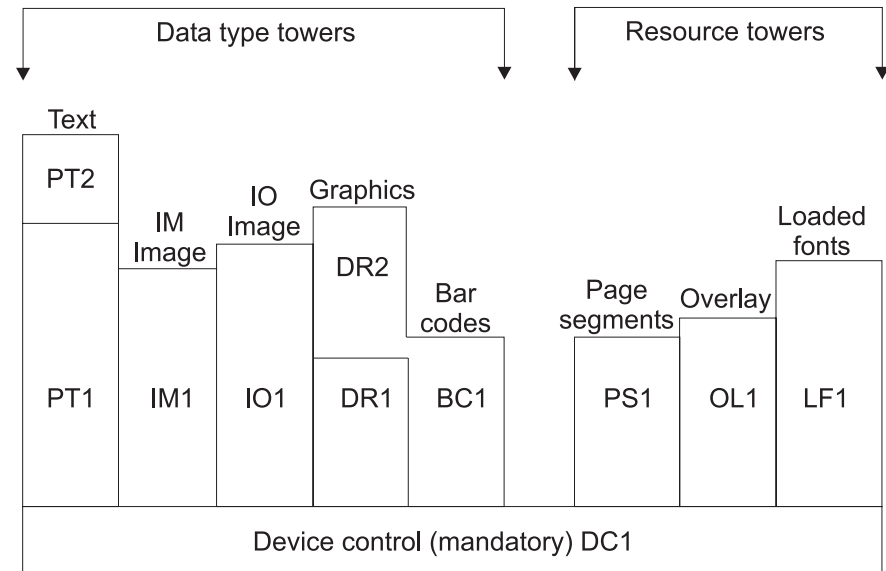
RBAFT531-0

Device control
Function set is composed of the IPDS commands that set up a page, communicate device controls, and manage acknowledgement protocol. The device-control function set is the only mandatory function set for IPDS printers, although not all DC1 orders are required to be supported.

Text Function set composed of commands and orders (text controls) required to present text information on a page, page segment (stored command sequence), or an overlay (electronic form). The text function set contains

two presentation text (PT) subsets: PT1 and PT2. Text printers support either subset. PT2 is a superset of PT1 and, therefore, all orders contained in PT1 are also in PT2.

IM image
Function set containing the IPDS commands required to present raster image data on a page, page segment, or an overlay.

IO image
Function set containing commands that present raster data (similar to IM image) but with additional functions.

Graphics
Function set composed of the IPDS commands and drawing orders required to present vector graphics on a page, page segment, or overlay. The graphics function set contains two drawing (DR) subsets: DR1 and DR2. Vector graphics printers support DR2, which is a superset of DR1.

Bar code
Function set composed of the IPDS commands required to present machine-readable bar code information on a page, page segment, or overlay.

Page segments and overlays
Function sets composed of the IPDS commands required to store and present IPDS constructs containing text, graphics, image, and bar code information. These stored constructs can be either page segments or overlays.

Loaded font
Function set composed of the IPDS commands necessary to load and delete font information.

Function-Set Requirements for IPDS
In order to claim support of the IPDS architecture, a product must do the following:

- Implement all required commands in the device-control function set
- Implement at least one subset of one other data function set
- Implement all required commands, orders, and controls for each supported function set or subset.

Return of Function-Set Information
A host presentation services program determines the functional capabilities of an IPDS printer by issuing certain IPDS query commands to the printer and requesting an acknowledgement. The data that the printer returns in the acknowledge reply shows the printer type and model, details of the function sets supported, and a variety of printer characteristics.

The IPDS Page Environment
IPDS creates mixed-data pages within a hierarchy of presentation spaces. These presentation spaces are: physical page, logical page, and data blocks.

Physical page
The medium (usually paper) on which information is placed. The physical page has boundaries of width and depth that define the limits of the medium.

Logical page
The electronic representation of the page that is sent to the printer. The

logical page is a rectangular area that may or may not be the same size as the physical page on which it is placed. Printing can only occur where the current logical page intersects the physical page (valid printable area).

Data blocks
Rectangular areas positioned on the logical page. Data blocks can be one of three types:

- Image Data Blocks: Raster information
- Graphics Data Blocks: Lines, curves, areas, and other drawing elements
- Bar Code Data Blocks: Bar-coded, machine-readable characters or human-readable characters.

> **Note:** There is no text data block. Text can be positioned anywhere in the valid printable area. Data blocks can be positioned in relation to the text.

IPDS has an hierarchical relationship between a physical page, a logical page, and data blocks. In the following figure, the logical page boundaries do not correspond to the physical page boundaries.

Physical page

Logical page

Text

Data block   *
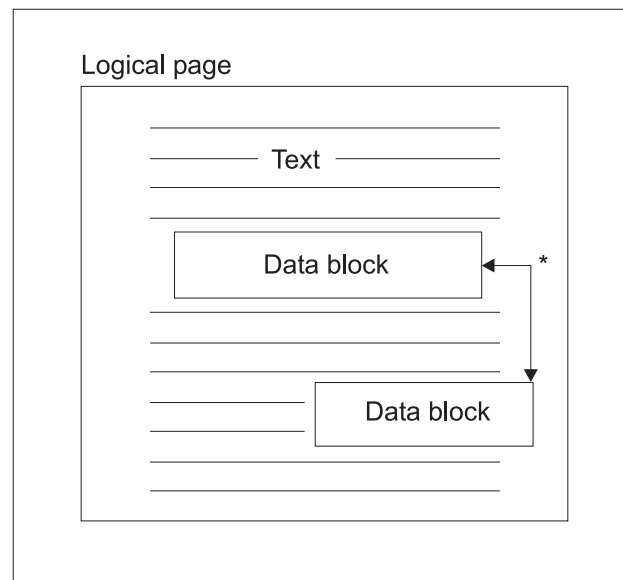
Data block

* Independent blocks of image, graphics, or bar code data

RBAFT532-0

One of the strengths of IPDS is that independent applications can create source data for each data block. The output of these independent applications is merged at the printer to create an integrated mixed data page. For example, text data could be produced on an editor like the OfficeVision editor, image data could be the output of a scanner stored in a folder, and graphics data could be produced by Business Graphics Utility. IPDS makes it possible to integrate application output rather than requiring the use of integrated applications.

Overlays and Page Segments
IPDS stores resources in the printer for later use. Overlays and page segments can be merged with the logical page before the logical page is printed on the physical page.

Overlay

Macro-like construct loaded by the host processor and sent to the printer's storage. An overlay may consist of any combination of text data, image block data, graphics block data, or bar code block data. An overlay contains the same type of presentation commands used in the logical page, but overlays are independent of the logical page environment. The major difference between overlays and logical pages is that overlays are stored until deleted but logical pages, if stored, are only stored until printed. Overlays are often used as electronic forms.

Page segment

This is like the overlay in makeup. The difference between a page segment and an overlay is that page segments are not independent of the page environment; they are merged with the logical page and assume the currently active environment.

Loaded Fonts

A font is a set of characters in a particular type style and size. Fonts can be downloaded from the host or may be resident in printer storage. Downloaded fonts are called *loaded fonts* and come in one of the following configurations:

Coded font

A complete code page of graphic characters in a particular style. (A code page maps each character in a font to a numeric value or code point.)

Symbol set

A set of characters simpler in structure than a coded font. Symbol sets are used where typographic quality is not required. Many dot matrix printers and displays use symbol sets.

Coordinate Systems

Xm, Ym Coordinate System (Physical Page)

IPDS uses orthogonal coordinate systems to define any point on a page. Distances between these coordinate systems are measured in logical units or *L-units* rather than physical pels.

The Xm, Ym coordinate system is the physical-page coordinate system. The top-left corner of the physical page is always (0,0).

The printer defines the top of a physical page.

RBAFT533-0
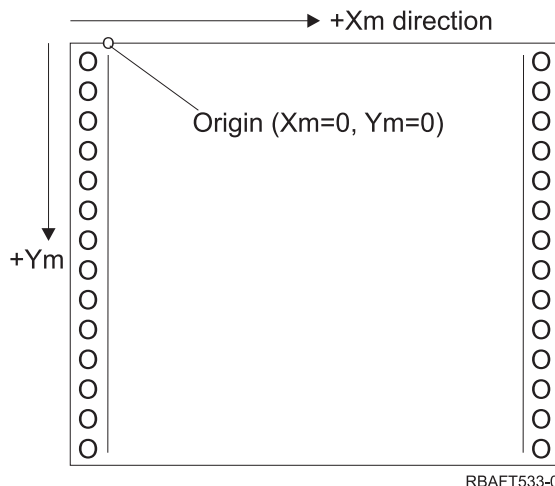
The Xm, Ym coordinate system is fixed for each media size. IPDS commands cannot change the orientation of these coordinates.

Xp, Yp Coordinate System (Logical Page)

The Xp, Yp coordinate system is the logical-page coordinate system. The origin of this system (Xp=0, Yp=0) is specified as an offset from the physical page origin (Xm=0, Ym=0) through the Load Page Position command. IPDS commands cannot change the orientation of the Xp, Yp coordinate system; it is always parallel to, but offset from, the Xm, Ym coordinate system.

The size of the logical page in the Xp dimension is called the *Xp extent*. The size of the logical page in the Yp dimension is called the *Yp extent*. The Xp, Yp coordinate system is used to locate data blocks on the logical page.

The coordinate system for overlays is the same as the Xp, Yp coordinate system for logical pages.

RBAFT534-0
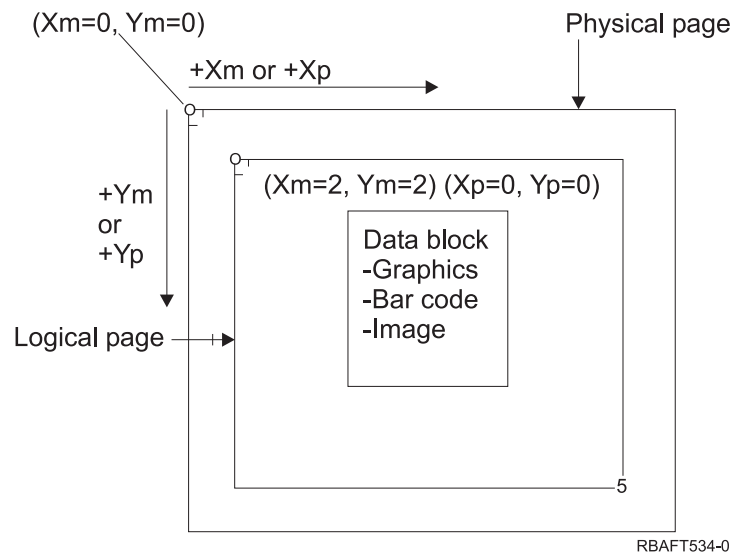
I, B Coordinate System (Text)

The Inline, Baseline (I, B) coordinate system describes the placement of data blocks on the logical page. The printer places characters along the I-axis to form a line of text and places lines of text along the B-axis on the logical page. IPDS commands can change both the origin and the orientation of the inline and baseline axes.

As characters are developed on the page, the inline coordinate is incremented in the **positive inline** (or +I) direction. As lines are developed on the page, the baseline coordinate is incremented in the **positive baseline** (or +B) direction.

Note: Characters are developed on a page in the direction in which they will be read (left to right, for example). The printer may actually place characters or lines on a page in various directions (as in bidirectional printing).

The coordinates of the first text position on the logical page are called the initial inline text coordinate (Io) and the initial baseline text coordinate (Bo). The coordinates of the current position on the logical page are called the current inline

text coordinate (Ic) and the current baseline text coordinate (Bc).


```
        ──────────►
         + I  Direction                    Logical  page
  + B │
      │        o  (Io,  Bo)
      │           Initial  text  position
      │
      ▼
               0  (Ic,  Bc)
                  Current  text  position
```

RBAFT535-0
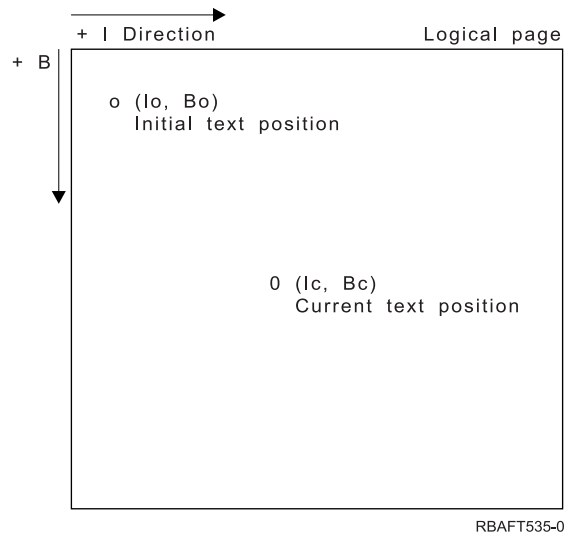
Processing IPDS Commands

The structured field format of IPDS allows commands to be sent to the printer in a continuous stream. Each command is self-describing. The command length, identifier, flag byte, and data (not always present) are all part of each command. The printer-host conversation is carried on as if IPDS commands were processed in sequential order by the printer.

Every IPDS command contains a flag byte. The setting on the acknowledgement-required bit on this flag byte indicates the end of a command sequence to the printer. The printer then sends an acknowledge reply to the host, as illustrated in the following diagram:


```
┌─────────────────────┐
│ IPDS command        │
└─────────────────────┘
         └──────────────►
           To printer

┌─────────────────────┐
│ IPDS command        │
└─────────────────────┘
         └──────────────►
           To printer

┌─────────────────────┐                    ┌─────────────────────┐
│ IPDS command        │                    │ Acknowledge reply   │
└─────────────────────┘                    └─────────────────────┘
         └──────────────►   ◄──────────────┘
           To printer          From printer to host
```

RBAFT536-0
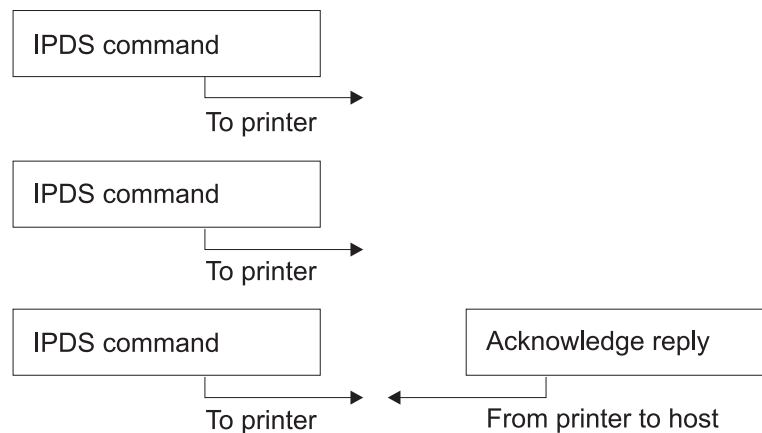
The IPDS Command Format

All IPDS commands are encoded in the following format:

Length	Command	Flag	CID	Data

Length

A 2-byte field that specifies the length of the command. This count

includes itself, the command field, the flag byte and the optional correlation ID (CID), and data fields. The length field can range from X'0005' to X'7FFF'.

Command
A 2-byte field that specifies the IPDS command.

Flag A 1-byte field that contains the IPDS command stream flags.
- Bit 0 is the acknowledgement required (ARQ) flag. If this bit is on, the host requests the printer to send an acknowledge reply.
- Bit 1 is the correlation ID (CID) flag. If it is on, a 2-byte correlation ID follows. If it is off, the CID is not present and the following bytes (if any) contain the data field.

CID (correlation ID)
A 2-byte field that specifies an identifier for the command. A presentation services program can use any value between X'0000' and X'FFFF' for the correlation ID.

Data Not present for all commands. If present, it contains specific orders, parameters, and data appropriate for the given command.

IPDS Operating States

IPDS commands are defined within the context of printer operating states. The printer moves between these operating states during command processing. IPDS printers are *state machines* with the following operating states:
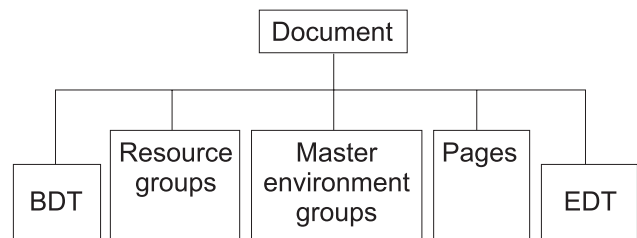
RBAFT537-0

- Home state
- Block state
  - IO image block state
  - IM image block state
  - Graphics block state
  - Bar code block state.
- Page state
- Overlay state
- Page segment state
- Font state
- Any-state

Home state
The initial IPDS operating state. The printer returns to home state at the end of each downloaded page, page segment, coded font, or overlay.

While in home state, the printer receives control and initialization commands to prepare for the print operation. In home state, the printer

can also receive commands that delete resources or request the return of printer information to the host presentation services program.

Block states

State for establishing the initial processing conditions for a block of data and placing the block of data on the logical page, page segment or overlay. The printer can only enter a block state from page, page segment, or overlay states.

Page state

The operating state for printing a logical page. The printer enters page state from home state on receiving a Begin Page command and exits on receiving an End Page command.

In page state, the printer can receive commands that merge previously defined and loaded overlays and page segments with the current page information. The printer can also receive Write Text commands that position text on the logical page and can enter a block state to write image, bar code, and graphics blocks.

Overlay state

State that allows overlay data to be stored in the printer. The printer enters overlay state from home state on receiving a Begin Overlay command and exits on receiving an End Page command.

In overlay state, the printer can receive commands that merge previously defined and loaded overlays and page segments with the current page information. The printer can also receive Write Text commands that position text on the logical page and can enter a block state to write image, bar code, and graphics blocks.

Page segment state

State that allows page segment data to be stored in the printer. The printer enters page segment state from home state on receiving a Begin Page Segment command and exits on an End Page command.

In page segment state, the printer can receive Write Text commands that position text on the logical page and can enter a block state to write image, bar code, and graphics blocks.

Font state

State that allows the printer to receive downloaded coded-font data. The printer enters font state from home state on receiving a Load Font Control command.

While the printer is in font state, the Load Font command can send coded-font, character-raster pattern data to the printer. Receipt of an End command returns the printer to home state.

Any-state

Some IPDS commands can be received in any IPDS operating state. These commands do not change the IPDS operating state, with the exception of XOA Discard Buffered Data.

Default Handling

Defaults are values used as control parameters when no other values are specified in the current command. IPDS defaults are invoked through omission or through values transmitted in the data field portion of commands. The IPDS default structure is normally hierarchical. General IPDS default rules are:

- If power has been interrupted or if the printer has been initialized, printer-established page default values are used until specific IPDS default values are received.
- Initial page values are established when the printer receives a Load Page Descriptor command. If no such command is received, printer-established default values remain in effect.
- Initial data block values are established when the printer receives either a Write Image Control, Write Image Control 2, Write Bar Code Control, or Write Graphics Control command. These values remain in effect until data controls override them or until the printer receives an End command that ends the block.

Mixed Object: Document Content Architecture (MO:DCA)

The ability to print documents with consistent output, independent of either operating system or printer, is extremely important to the user of printed data. In order to help achieve this goal, IBM has defined a single object-oriented data stream—**Mixed Object Document Content Architecture (MO:DCA)**. (An object is a collection of data that can be treated as a unit.) This architecture has been developed in order to meet several objectives:

- The requirements relating to document and data sharing specified in IBM's Systems Application Architecture
- Co-existence and migration of existing IBM document architecture and printer data streams
- Device independence
- Separation of functions to simplify transformation of objects into other data streams
- National Language Support
- Office Document Architecture (ODA) support
- Standard Generalized Markup Language (SGML)

The strategic architecture for the interchange of revisable and presentation form of documents and objects used as resources is MO:DCA, which has evolved from Revisable Form Text: Document Content Architecture (RFT:DCA).

The data stream for an MO:DCA document consists of various objects, such as text, images, and graphics, as well as the logical and layout structure of the document. The logical structure defines the logical content of the document—chapters, figures, and lists. The layout structure defines the way the data should be presented.
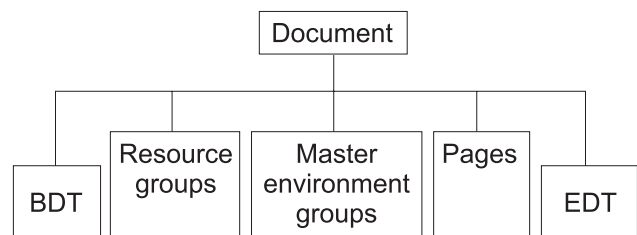
RBAFT537-0

Figure 8. MO:DCA Document Structure

BDT (Begin document)
> Indicates the beginning of the document

Resource groups
Specifies fonts, overlays, and segments so that these objects can be transmitted as part of the data stream. They can be referenced by an MO:DCA Include structured field.

Master environment groups
Specifies the processing environment, such as space definitions, suppression of data, number of copies, and internal data stream references.

Pages Contains objects that are part of the document. These objects could be text, graphics, and images.

EDT (End document)
Indicates the end of the document.

The following different types of objects make up MO:DCA. All of these objects are supported by IPDS:
- Bar Code Object Content Architecture (BCOCA)
- Image Object Content Architecture (IOCA)
- Graphics Object Content Architecture (GOCA)
- Presentation Text Object Content Architecture (PTOCA)
- Font Object Content Architecture (FOCA).

Bar Code Object Content Architecture (BCOCA)

A bar-code object could contain "draw rule" commands or raster data, depending on whether the bar code is to be drawn as a graphics object or has been scanned into the data stream as an image. A bar code object containing draw rule commands is built up using only lines of a specified length and width. A graphics object is constructed from a number of primitives, such as lines, arcs, symbols, shaded areas, and point arrays.

Image Object Content Architecture (IOCA)

IOCA represents images in device-independent format. A standard set of constructs has been defined to describe the image data, the characteristics of that data, and manipulation functions that may be performed on the data. The image content is inserted in an image segment.

Graphics Object Content Architecture (GOCA)

GOCA describes complex pictures. These pictures are formed from a collection of primitives, such as lines, arcs, characters, symbols, and shaded areas or point arrays. Each of these primitives has its own set of attributes, such as line width, orientation, and resolution. In addition to these attributes, there is a set of general drawing attributes like color, which apply to all primitives.

Presentation Text Object Content Architecture (PTOCA)

PTOCA describes the text part of a document. The presentation text object, in common with the other objects, is designed not only to be carried by, but to be an integral part of, the data stream, providing the following:
- Structured field introducer and syntax for the structured field
- Begin/end object structure
- Control of alternate action selection for error recovery
- Passing of exception conditions back to the originating process
- Initial state of the object
- Relationship of presentation text objects to other objects contained in the data stream.

Two structured fields provide the necessary presentation information to the printer:

P T descriptor structured field
> Defines several positional parameters for the object

P T data structured field
> Contains the presentation text and the control sequences for positioning graphic characters. These graphic characters are defined within the coded fonts.

Font Object Content Architecture (FOCA)

In order to achieve uniform document presentation output, it is essential that font resources be consistently defined and implemented. These resources must be identified by means of a constant, unvarying set of parameters.

FOCA makes it possible to achieve the required degree of consistency by defining:

- A common font and character definition model that can be used by all products and architectures as the basis for font applications
- A composite set of parameters specific to a font resource and references to that resource
- A device-and-technology-independent method of defining font measurements
- The specification of formats for conveying font information to suit the application

FOCA defines the parameter content of:

- IBM font resources
- References to the font resources
- Information accessed by the font resources

American National Standard Code for Information Interchange (ASCII)

There is no formal structure controlling the use of the ASCII data stream to control printers attached to systems providing ASCII support. Control of page printers, like the IBM 3812, is exercised using page map primitives (PMPs), which are a set of commands or basic instruction set of these printers when attached in ASCII mode. ASCII data sent to a page printer is translated into PMPs. The page printer composes the page of data in its internal memory or page map. Two page orientations (portrait and landscape) as well as four print directions are supported. Complexity of the printed data is determined by the application print program, which can set the pels on explicitly in the page set, or implicitly, by instructing the printer to generate characters or vectors (lines). Fonts available for printing are stored on the printer's microcode/font diskette. Most page printers support **macros**, which are a saved list of PMP commands, avoiding the necessity for the application program to send down a string of individual commands each time a particular printed function is required.

There are five basic categories of PMP commands:

Page commands
> Set overall page parameters, such as size and orientation

Cursor commands
> Move the cursor on the page map

Font commands
> Manage fonts within the page printer

Generation commands
Create pels on the page map

Macro commands
Allow strings of other commands to be saved for later processing.

Printing capabilities and functions in ASCII attach mode are governed by individual application programs that are written to suit the capabilities of specific printers (or printers that provide an emulation of that printer). There is no architectural data stream standard to which ASCII printers can conform in the interests of uniformity. ASCII printing applications are therefore totally printer dependent.

On OS/400, ASCII printing support is provided by translating native EBCDIC characters to the ASCII equivalents.

PostScript

PostScript is not a data stream but rather a page description language developed by Adobe Systems Incorporated and used to prepare page layouts for printing. This involves not only text with typesetting information on sizes and fonts, but graphics and scanned images as well. An example is Aldus Pagemaker**, which produces PostScript for downloading to the IBM personal pageprinter adapter. This adapter contains the PostScript Interpreter developed for IBM by Adobe Systems Incorporated. The Interpreter produces the video image for the IBM Personal Pageprinter 4216 Model 20, 4019, or 4029 with the postscript feature.

The following licensed programs produce PostScript output:
- IBM Personal Publishing System
- IBM Interleaf Publishing Series
- Document Composition Facility (DCF) Release 3.2
- Image Handling Facility (IHF) Release 1.2.

There is currently no support for PostScript on the iSeries server.

Appendix F. Double-Byte Character Set Support

This appendix contains information that you need if you use double-byte characters. This includes the following topics:

- Double-byte character set (DBCS) fundamentals
- Processing double-byte characters
- Device file support
- Printer support
- Spooling support

Other DBCS device file support and conversion information can be found in the File Management topic in the iSeries Information Center.

Double-Byte Character Set Fundamentals

Some languages, such as Chinese, Japanese, and Korean, have a writing scheme that uses many different characters that cannot be represented with single-byte codes. To create coded character sets for such languages, the system uses 2 bytes to represent each character. Characters that are encoded in 2-byte code are called double-byte characters. **A double-byte character set** (DBCS) is a set of characters in which each character is represented by 2 bytes.

The following example shows alphanumeric characters coded in a single-byte code scheme and double-byte characters coded in a double-byte code scheme.

```
          1-Byte Code                      2-Byte Code
(SBCS)                          (DBCS)
  A  ──── X'C1'                    A  ──── X'42C1'
  B  ──── X'C2'                    B  ──── X'42C2'
  1  ──── X'F1'                    1  ──── X'42F1'
  2  ──── X'F2'                    2  ──── X'42F2'
                                   あ ──── X'4481'
                                   美 ──── X'457D'   ( Japanese )
                                   橋 ──── X'8877'   ( Japanese )
                                   강 ──── X'525F'   ( Korean )
                                   进 ──── X'4F99'   ( Simplified Chinese )
                                   進 ──── X'5B70'   ( Traditional Chinese )

        X'hhhh' indicates that the code has the hexadecimal value, "hhhh".

1-Byte Codes:   │ 256 characters │

2-Byte Codes:   │ 256 X 256 characters                          │
                                                          HRSLS338-2
```

You can use double-byte characters as well as single-byte characters in one application. For instance, you may want to store double-byte data and single-byte data in your database, create your display screens with double-byte text and fields, or print reports with double-byte characters.

DBCS Code Scheme

IBM supports two DBCS code schemes: one for the host system, the other for personal computers. The IBM-host code scheme has the following code-range characteristics:

First byte
> hex 41 to hex FE

Second byte
> hex 41 to hex FE

Double-byte blank
> hex 4040

In the following figure, using the first byte as the vertical axis and the second byte as the horizontal axis, 256 x 256 intersections (code points) are expressed. The lower-right code area is designated as the valid double-byte code area and X is assigned to the double-byte blank.

D: double-byte code area
x: double-byte blank
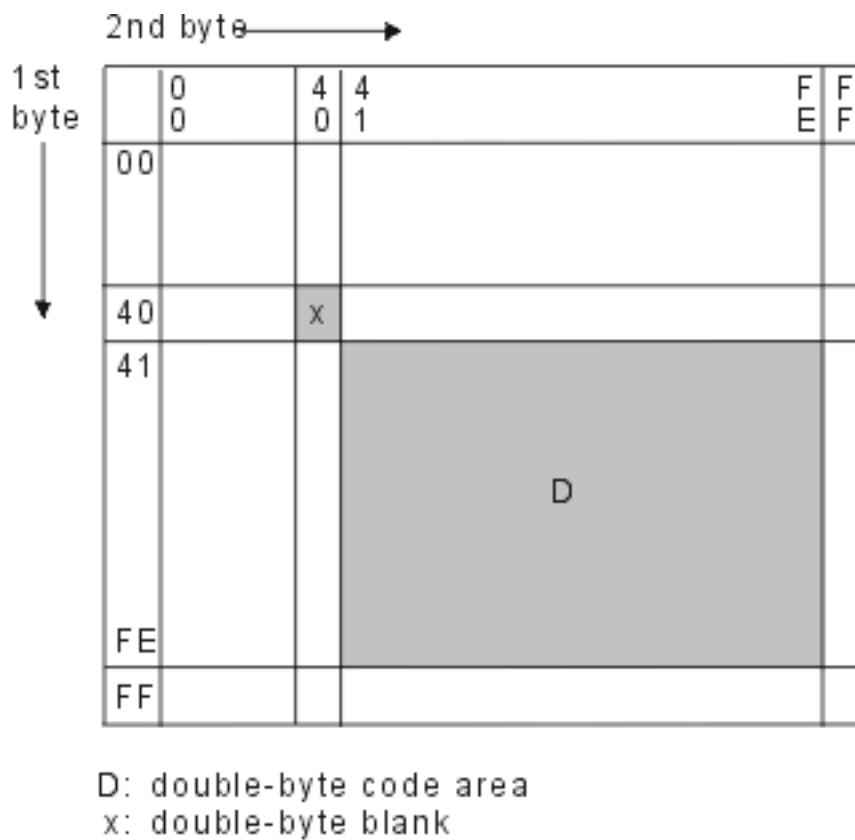
Figure 9. IBM-Host Code Scheme

By assigning the values hex 41 to hex FE in the first and second bytes as the DBCS character codes, the codes can be grouped in wards with 192 code points in each ward. For example, the code group with the first byte starting with hex 42 is called *ward 42*. Ward 42 has the same alphanumeric characters as those in a corresponding single-byte EBCDIC code page, but with double-byte codes. For example, the character *A* is represented in single-byte EBCDIC code as hex C1 and in IBM-host code as hex 42C1.

The iSeries server supports the following double-byte character sets:

- IBM Japanese Character Set
- IBM Korean Character Set
- IBM Simplified Chinese Character Set
- IBM Traditional Chinese Character Set

The following tables show the code ranges for each character set and the number of characters supported in each character set.

Table 48. IBM Japanese Character Set

Wards	Content	Number of Characters
40	Space in 4040	1
41 to 44	Non-Kanji characters • Greek, Russian, Roman numeric (Ward 41) • Alphanumeric and related symbols (Ward 42) • Katakana, Hiragana, and special symbols (Ward 43-44)	549
45 to 55	Basic Kanji characters	3226
56 to 68	Extended Kanji characters	3487
69 to 7F	User-defined characters	Up to 4370
80 to FE	Reserved	
: Total number of IBM-defined characters: 7263		

Table 49. IBM Korean Character Set

Wards	Content	Number of Characters
40	Space in 4040	1
41 to 46	Non-Hangeul/Hanja characters (Latin alphabet, Greek, Roman, Japanese Kana, numeric, special symbols)	939
47 to 4F	Reserved	
50 to 6C	Hanja characters	5265
6D to 83	Reserved	
84 to D3	Hangeul characters (Jamo included)	2672
D4 to DD	User-defined characters	Up to 1880
DE to FE	Reserved	
: Total number of IBM-defined characters: 8877		

Table 50. IBM Simplified Chinese Character Set

Wards	Content	Number of Characters
40	Space in 4040	1

Table 50. IBM Simplified Chinese Character Set (continued)

Wards	Content	Number of Characters
41 to 47	Non-Chinese characters (Latin alphabet, Greek, Russian, Japanese Kana, numeric, special symbols)	712
48 to 6F	Chinese characters: Level 1 and Level 2	3755 and 3008
70 to 75	Reserved	
76 to 7F	User-defined characters	Up to 1880
80 to FE	Reserved	
: Total number of IBM-defined characters: 7476		

Table 51. IBM Traditional Chinese Character Set

Wards	Content	Number of Characters
40	Space in 4040	1
41 to 46	Non-Chinese characters (Latin alphabet, Greek, Roman, Japanese Kana, numeric, special symbols)	674
47 to 4B	Reserved	
4C to 68	Primary Chinese characters	5401
69 to 91	Secondary Chinese characters	7652
92 to CF	Reserved	
D0 to DD	User-defined characters	Up to 6204
DE to FE	Reserved	
: Total number of IBM-defined characters: 13728		

This code scheme applies to the iSeries server, System/36, and System/38, as well as the System/390* system. A different DBCS code scheme, called the IBM Personal Computer DBCS code scheme, is used on the Personal System/55. For details of the IBM Personal Computer DBCS code scheme, refer to IBM PS/55 publications.

Shift-Control Characters

When the IBM-host code scheme is used, the system sometimes uses shift-control characters to identify the beginning and end of a string of double-byte characters. The shift-out (SO) character, hex 0E, indicates the beginning of a double-byte character string. The shift-in (SI) character, hex 0F, indicates the end of a double-byte character string.
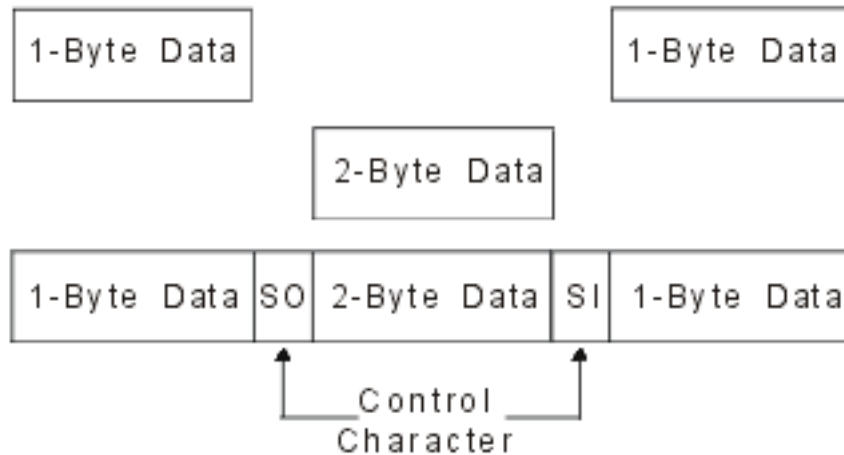
Each shift-control character occupies the same amount of space as one alphanumeric character. By contrast, double-byte characters occupy the same amount of space as two alphanumeric characters.

Printer files created with DEVTYPE(*AFPDS) do not have to use SO and SI. Instead the user may change to a DBCS font. The IPDS printer recognizes that the font is DBCS, and processes the user data accordingly.

Invalid Double-Byte Code and Undefined Double-Byte Code

Invalid double-byte code has a double-byte code value that is not in the valid double-byte code range. Figure 9 on page 500 shows valid double-byte code ranges. This is in contrast to undefined double-byte code where the double-byte code is valid, but no graphic symbol has been defined for the code.

Using Double-Byte Data

This section tells you where you can use double-byte data and discusses the limitations to its use.

Where You Can Use

You can use double-byte data in the following ways:

- As data in files:
  - Data in database files.
  - Data entered in input-capable and data displayed in output-capable fields of display files.
  - Data printed in output-capable fields in printer files.
  - Data used as literals in display files and printer files.
- As the text of messages.
- As the text of object descriptions.
- As literals and constants, and as data to be processed by high-level language programs.

Double-byte data can be displayed only at DBCS display stations and printed only on DBCS printers. Double-byte data can be written onto diskettes and tape, as well as onto disk storage.

Where You Cannot Use

You cannot use double-byte data in the following ways:

- As OS/400 object names.
- As command names or variable names in control language (CL) and other high-level languages.
- As displayed or printed output on alphanumeric work stations.

Double-Byte Character Size

When displayed or printed, double-byte characters are twice as wide as alphanumeric characters.

Consider the width of double-byte characters when you calculate the length of a double-byte data field because field lengths are usually identified as the number of alphanumeric character positions used. For more information on calculating the length of fields containing double-byte data, refer to the DDS Reference topic in the online Information Center.

Processing Double-Byte Characters

Due to the large number of double-byte characters, the system needs more information to identify each double-byte character than is needed to identify each alphanumeric character.

There are two types of double-byte characters: basic and extended. These characters are usually processed by the device on which the characters are displayed or printed.

Note: This does not apply to IPDS printers that support DBCS data. For IPDS printers, all characters printed are downloaded from the iSeries server.

Basic Characters

A DBCS device can process basic double-byte characters without any assistance from the system. The device knows about the graphic characters because they are stored in the device. The number of double-byte characters that are stored in the device varies with the language supported and the storage size of the device. A DBCS device can display or print basic characters without using the extended character processing function of the operating system.

Extended Characters

When processing extended characters, the device requires the assistance of the system. The system must tell the device what the character looks like before the device can display or print the character. Extended characters are stored in a DBCS font table, not in the DBCS device. When displaying or printing extended characters, the device receives them from the DBCS font table under control of the operating system.

Extended character processing is a function of the operating system that is required to make characters stored in a DBCS font table available to a DBCS device.

To request extended character processing, specify the double-byte extended character parameter, IGCEXNCHR(*YES), on the file creation command when you create a display (CRTDSPF command) or printer file (CRTPRTF command) that processes double-byte data. Because IGCEXNCHR(*YES) is the default value, the system automatically processes extended characters unless you instruct it

otherwise. You can change this file attribute by using a change file (CHGDSPF or CHGPRTF) or override file (OVRDSPF or OVRPRTF) command. For example, to override the display file DBCSDSPF so that extended characters are processed, enter:

```
OVRDSPF DSPF(DBCSDSPF) IGCEXNCHR(*YES)
```

Notes:

1. The system ignores the IGCEXNCHR parameter when processing alphanumeric files.

2. When you use the Japanese 5583 Printer to print extended characters, you must use the Kanji print function of the Advanced DBCS Printer Support licensed program. Refer to *AS/400 Utilities: Kanji Print Function User's Guide and Reference* , SH18-2179, for how to use this utility.

What Happens When Extended Characters Are Not Processed

When extended characters are not processed:

- Basic double-byte characters are displayed and printed.
- On displays, the system displays the undefined character where it would otherwise display extended characters.
- On printed output, the system prints the undefined character where it would otherwise print extended characters.
- The extended characters, though not displayed or printed, are stored correctly in the system.

Device File Support

The following sections describe DBCS device files and special considerations for working with DBCS device files. Data description specifications (DDS), a language used to describe files, can be used with DBCS device files. For information about using DDS, refer to the DDS Reference topic in the iSeries Information Center.

What a DBCS File Is

A *DBCS file* is a file that contains double-byte data or is used to process double-byte data. Other files are called *alphanumeric files*.

The following types of device files can be DBCS files:

- Display
- Printer
- Tape
- Diskette
- ICF

When to Indicate a DBCS File

You should indicate that a file is DBCS in any of the following situations:

- The file receives input, or displays or prints output, that has double-byte characters.
- The file contains double-byte literals.
- The file has double-byte literals in the DDS that are used in the file at processing time (such as constant fields and error messages).
- The DDS of the file includes DBCS keywords. See the DDS Reference topic in the online Information Center for information on these keywords.

- The file stores double-byte data (database files).

How to Indicate a DBCS File

You must indicate that a device file is a DBCS file in order for the system to process double-byte data properly. You can do this in one of the following ways:

- Through DDS
  - DDS provides fields of the following data types.
    - DBCS-Only Field (Type J)

      DBCS-only fields display and accept only double-byte characters. Double-byte characters are always enclosed in shift-out and shift-in characters that have to be paired.
    - Open Field (Type O)

      Open fields display and accept both single-byte and double-byte characters. Double-byte characters are enclosed in shift-out and shift-in characters that have to be paired.
    - Either Field (Type E)

      Either fields display and accept *either* single-byte or double-byte characters, but not *both*. Double-byte characters are enclosed in shift-out and shift-in character pairs.
    - Graphic Field (Type G)

      Graphic fields display and accept only double-byte characters. The double byte characters are *not* enclosed in shift-out and shift-in pairs.
  - In printer files, by defining fields with DBCS-graphic data type (Type G).
  - In printer and ICF files, by defining fields with DBCS open data type (type O).
  - In display files, by defining fields with DBCS-only data type (type J), either data type (type E), or open data type (type O).
  - By using a double-byte literal that is used with the file at processing time, such as literals specified with the Default (DFT) and Error Message (ERRMSG) DDS keywords.

    Note: You may also use double-byte literals as text and comments in a file, such as with the DDS keyword TEXT. However, the system does not consider a file, whose only DBCS attribute is that it has double-byte comments, to be a DBCS file.

  - By specifying the Alternative Data Type (IGCALTTYP) DDS keyword in display and printer files. This keyword lets you use display and printer files with both alphanumeric and double-byte applications. When you put the IGCALTTYP keyword into effect, you can use double-byte data with the file.

    Put the IGCALTTYP keyword into effect by creating, changing, or overriding display and printer files with the IGCDTA(*YES) value. You can put the IGCALTTYP keyword into effect for display and printer files by specifying IGCDTA(*YES) on the following device file commands:
    - Create Display File (CRTDSPF)
    - Create Printer File (CRTPRTF)
    - Change Display File (CHGDSPF)
    - Change Printer File (CHGPRTF)
    - Override with Display File (OVRDSPF)
    - Override with Printer File (OVRPRTF)

When you specify IGCDTA(*NO), the IGCALTTYP keyword is not in effect and you can use only alphanumeric data with the file. Changing or overriding the file to put the IGCALTTYP keyword into effect does not change the DDS of the file.

Except when using the IGCALTTYP function, you do not need to specify IGCDTA(*YES) on the file creation command if you have already specified DBCS functions in the DDS. Instead, specify IGCDTA(*YES) when the file has DBCS functions that are not indicated in the DDS. For example, specify IGCDTA(*YES) on the file creation command if the file is intended to contain double-byte data.

- By specifying IGCDTA(*YES) on the following device file creation commands:
  - Create Diskette File (CRTDKTF)
  - Create Display File (CRTDSPF)
  - Create Printer File (CRTPRTF)
  - Create Tape File (CRTTAPF)
- By specifying IGCDTA(*YES) on the following database file creation commands:
  - Create Physical File (CRTPF)
  - Create Source Physical File (CRTSRCPF)

Note: DBCS-graphic data type fields are supported for externally-described (DDS) printer files only. For program-described printer files, the application program must enclose the DBCS-graphic data type fields with the appropriate shift-out and shift-in characters.

Improperly Indicated DBCS Files

If you do not properly indicate that a file is a DBCS file, one of the following happens:

- For printer files, printer data management assumes the output data to the printer does not contain double-byte data. The end result depends on the type of printer the data is printed on and the status of the replace unprintable character parameter for the printer file you are using.

  If the replace-unprintable-character option is selected, printer data management interprets shift-control characters as unprintable characters and replaces them with blanks. The double-byte data itself is interpreted as alphanumeric data, and the printer attempts to print it as such. The printed double-byte data does not make sense.

  If the replace-unprintable-character option is not selected and the printer is an alphanumeric printer, the double-byte data, including the control characters, is sent as-is to the printer. On most alphanumeric printers, the shift-control characters are not supported, and an error occurs at the printer.

  If the replace-unprintable-character option is not selected and the printer is a DBCS printer, the double-byte data is printed with the exception of extended characters. Because the file was not indicated as a DBCS file, the system does not perform extended character processing. The extended characters are printed with the symbol for undefined double-byte characters.

- For display files, display data management assumes that the output data to the display does not contain double-byte data. The end result depends on whether the display is an alphanumeric or DBCS display.

  If the display is an alphanumeric display, the double-byte data is interpreted as alphanumeric data. The shift-control characters appear as blanks. The displayed double-byte data does not make sense.

If the display is a DBCS display, the double-byte data is displayed with the exception of extended characters. The system does not perform extended character processing on the data. Therefore, extended characters are displayed with the symbol for undefined double-byte characters.

- The system does not recognize literals with DBCS text as double-byte literals if the source file is not specified as a DBCS file.

Making Printer Files Capable of DBCS

In many cases, printer files are used by the system to produce data that will eventually be printed or displayed. In these cases, the data is first placed into a spooled file using one of the IBM-supplied printer files. The data is then taken from the spooled file and is displayed or printed based on the request of the user.

When the data involved contains double-byte characters, the printer file that is used to place the data into the spooled file must be capable of processing double-byte data. A printer file is capable of processing double-byte data when *YES is specified on the IGCDTA parameter for the file. In most cases, the system recognizes the occurrence of double-byte data and takes appropriate measures to ensure the printer file that is used is capable of processing double-byte data.

In some cases, however, the system cannot recognize the occurrence of double-byte data and may attempt to use a printer file that is not capable of processing double-byte data. If this occurs, the output at the display or printer may not be readable. This can happen when object descriptions containing double-byte characters are to be displayed or printed on an alphanumeric device.

To ensure that you receive correct results when you display or print double-byte characters, some recommendations should be followed. Action is required on your part if you have a single-byte national language installed as a secondary language. Printer files that are received as part of the DBCS version of a product are always capable of processing DBCS data.

The following recommended actions should be performed after the product or feature has been installed:

1. If all printers and display devices attached to your system are DBCS-capable, you can enable all printer files for double-byte data. For IBM-supplied printer files that are received as part of a single-byte secondary language feature, you can enable all printer files by issuing the following command:

   CHGPRTF FILE(*ALL/*ALL) IGCDTA(*YES)

   Note: The IBM-supplied printer file QPSPLPRT should **not** have the IGCDTA parameter value changed to *YES. If you run the CHGPRTF command in the above example you must change the IGCDTA parameter value back to *NO.

   After this command has been completed, all printer files in all libraries will be enabled for double-byte data. The change will be a permanent change.

2. If all printer and display devices attached to your system are not DBCS-capable, it is recommended that you do not enable all IBM-supplied printer files.

   Instead, use the library search capabilities of the system to control which printer files will be used for any particular job. When the potential exists that double-byte data will be encountered, the library list for the job should be such that the printer files that are DBCS-enabled will be found first in the library list.

Conversely, if only single-byte data is expected to be encountered, the library list should be set up so the printer files that are not enabled for DBCS will be found first. In this way, the printer file capabilities will match the type of data that will be processed. The decision as to what type of printer file to use is made on the basis of what type of data will be processed. The device that will be used to actually display or print the data may also influence this decision.

In some cases it may be desirable to make the printer file only temporarily DBCS-capable instead of making a permanent change. For a specific job, you can make this temporary change by using the OVRPRTF command.

To temporarily enable a specific printer file, you can use the following command:

```
OVRPRTF FILE(filename) IGCDTA(*YES)
```

where `filename` is the name of the printer file you want to enable.

Printer Support

You should be familiar with both the "Device File Support" on page 505 and DDS for DBCS printer files before reading this section.

Special DBCS Printer Functions

The DBCS printers offer the following functions:

- Character rotation
- Character expansion
- Condensed printing
- Shift-control character printing

Character Rotation

The DBCS printers can rotate double-byte characters 90 degrees counterclockwise before printing so that the printed output can be read vertically.

For example, the character rotation function takes characters as shown:

文字を旋回する

HRSLS302-2

and rotates them so that you can read the printed characters vertically:

文字を旋回する

HRSLS303-2

Specify character rotation with the IGCCHRRTT parameter on the Create Printer File (CRTPRTF), Change Printer File (CHGPRTF), and Override with Printer File (OVRPRTF) commands, or with the IGCCHRRTT keyword in the DDS for the file you are printing. This function rotates only double-byte characters. It does not rotate alphanumeric characters.

Character Expansion (SCS DBCS Printers Only)

The DBCS printers can expand characters to twice their normal width or their normal height. Specify the character expansion with the DDS character size (CHRSIZ) keyword. For example, if you specify the value CHRSIZ(2 1), the following characters:

文字を横倍角にする

HRSLS304-2

are printed twice as wide, but the height remains the same.

文字を横倍角にする

HRSLS305-2

To print twice as wide <u>and</u> twice as high, you would specify CHRSIZE (2 2).

Condensed Printing (SCS DBCS Printers Only)

The DBCS printers can print 20 double-byte characters per 3 inches so that more double-byte characters fit on a printed line. For example, the following characters shown:

when condensed, are printed as:

文字の密度を変更する

HRSLS306-2

文字の密度を変更する

HRSLS307-2

Specify condensed character printing with the IGCCPI parameter on the CRTPRTF, CHGPRTF, and OVRPRTF commands.

Defining a Line (SCS DBCS Printers Only)

The record-level define line (DFNLIN) keyword in DDS can be used to draw a horizontal or vertical line (also known as a grid line). A horizontal line is drawn at the bottom of the character spaces. A vertical line is drawn on the left edge of the character spaces. You can draw horizontal lines and vertical lines to form boxes on the printed output.

The DFNLIN keyword is valid for SCS printers.

The maximum number of lines that can be printed at one time is 200. The maximum number of active vertical lines (vertical lines currently being printed on the page) is 150. More than 200 DFNLIN keywords may be used per page if all the define lines from the previous records have been printed.

Output considerations at run time:

- Spacing and skipping are processed before the DFNLIN keyword. If you space or skip past the start of a line, that line will be truncated (or not printed if the end of the line is passed also).
- A horizontal line cannot extend over a page boundary. A horizontal or vertical line cannot be started over a page boundary.
- The start line value specified on the DFNLIN keyword cannot be larger than the page length value specified on the PAGESIZE parameter on the printer.
- The start position value specified on the DFNLIN keyword cannot be larger than the page width value specified on the PAGESIZE parameter.
- The sum of the length and the start line value for a vertical line (specified on the DFNLIN keyword) cannot be larger than the page length specified on the PAGESIZE parameter.

- The sum of the length and the start position value for a horizontal line (specified on the DFNLIN keyword) cannot be larger than the page width specified on the PAGESIZE parameter.

A diagnostic message is sent whenever the PAGESIZE and DFNLIN values together cannot correctly process a request.

The following is an example of using DFNLIN to produce lines in a table:

社　員　番　号	氏　　　名
010001	山田一郎
010002	日本一郎

HRSLS308-2

Shift-Control Character Printing

The DBCS printers can print shift-control characters in one of the following ways:
- Suppress the shift-control characters so that these characters do not occupy any space on printed output.
- Print one blank in the space occupied by each shift-control character.
- Print two blanks in the space occupied by the shift-in character and suppress the shift-out character.

Specify how to print shift-control characters on the DBCS printers with the IGCSOSI parameter on the CRTPRTF, CHGPRTF, and OVRPRTF commands.

For data printed using the DBCS-graphic data type with an externally described printer file, shift-out/shift-in processing is not used. Instead, the shift control characters added to the DBCS data do not occupy any space on the printed output.

Double-Byte Character Printing Considerations

When you print double-byte data, consider the following:
- Extended character printing
- Condensed printing
- Unprintable double-byte characters
- Double-byte characters in an alphanumeric field
- Spanned lines
- Spanned pages
- Use of the Print key
- End-of-forms on the 5553 Printer
- Double-byte characters printed on alphanumeric printers

Extended Character Printing

Specify extended character processing to make sure that extended characters are processed. Otherwise, the system prints only basic double-byte characters. See "Processing Double-Byte Characters" on page 504 for instructions on specifying extended character processing and for information on the effects of such processing.

Condensed Printing

When specifying condensed printing on DBCS printers (by specifying IGCCPI(*CONDENSED) on the CRTPRTF, CHGPRTF, or OVRPRTF command), consider the following:

- Specify the page width in alphanumeric print positions with the CPI parameter. Although the record to be printed may contain 88 double-byte characters (which would use 176 print positions in normal printing) and the page width is 132 print positions, the double-byte data should print properly in condensed mode.

- For program-described printer files, data might not be printed in the proper position on the page. The system does not perform boundary alignment for alphanumeric data in printed records. When double-byte and alphanumeric data are printed on the same line, the printer begins printing alphanumeric data in the first space following the double-byte data. As a result, characters might not be printed on the proper position on the page.

- For DDS files, the printer begins printing alphanumeric data in the first position following the double-byte data, when double-byte and alphanumeric characters are mixed in a field defined with data type O (double-byte-capable). As a result, data might not be printed on the proper position on the page. This situation does not arise when the field contains only double-byte data or when alphanumeric data is printed in a field defined with an alphanumeric data type.

Selecting the Appropriate Page Width

Page width is specified as the second value of the PAGESIZE parameter on the CRTPRTF, CHGPRTF, or OVRPRTF commands. The correct page width depends on the printer being used and the characters per inch (CPI) specified for the printer file.

When describing printer files used with printers configured as a 5553 Printer, select a page size in the range based on characters per inch:

CPI	Page-Width Range
10	1 through 136
12	1 through 163
13.3	1 through 181
15	1 through 204
18	1 through 244
20	1 through 272

Choose one of the following (depending on the CPI selected) when describing printer files used with printers configured as a 5583 Printer:

CPI	Page-Width Range
10	1 through 132
12	1 through 158
13.3	1 through 176
15	1 through 198
18	1 through 236
20	1 through 264

Unprintable Double-Byte Characters

A double-byte character is considered unprintable if its double-byte code is not in the valid range or if its double-byte code is valid but does not have a character image defined.

You can specify that the system replace unprintable double-byte characters by specifying the replace unprintable character parameter (RPLUNPRT(*YES)) on the CRTPRTF, CHGPRTF, or OVRPRTF command, but you cannot choose the replacement character.

Although you cannot choose the replacement character for unprintable double-byte characters, you can choose the replacement character for unprintable alphanumeric characters. To improve system performance, select a blank ( ) as the replacement character for unprintable alphanumeric characters.

When the system finds an unprintable double-byte character during printing, the following happens:

- If you specify RPLUNPRT(*YES), the system does not send a message when it finds unprintable characters. Instead, the system prints unprintable extended characters as either the double-byte underline (__) when you specify extended character processing, or as an undefined character when you do not specify extended character processing.

  For Japanese printers, the default symbol used is:

  For Chinese and Korean printers, the default symbol used is the underscore.

  The system prints unprintable basic double-byte characters as double-byte blanks.

- If you specify RPLUNPRT(*NO), the system sends an inquiry message when it finds unprintable characters. You have the following options:

  - Hold the spooled file.

  - Continue printing where the unprintable character was encountered. If you continue printing, the system sends the inquiry message that you just received. It is sent each time the system finds an unprintable character, regardless of your response to the first message.

  - Continue printing by specifying a page number where printing should continue. When the system finds subsequent unprintable characters, it processes the characters as if the file were specified with RPLUNPRT(*YES). See the item in this list about RPLUNPRT(*YES) for a description of how the system processes these characters.

If the system finds invalid double-byte code, it stops processing double-byte extended characters and prints them as the undefined character.

Double-Byte Data in an Alphanumeric Field

If you try to print double-byte data in a field that is described in DDS as alphanumeric, the system interprets the data as alphanumeric. What happens depends on whether the printer being used is an alphanumeric or DBCS printer, and on the status of the replace-unprintable-characters option. This condition is a special case described under "Improperly Indicated DBCS Files" on page 507.

Spanned Lines

If a printed line of double-byte data exceeds its specified page width (line length), the system tries to continue printing the data. To do this, the system ignores the FOLD parameter on the CRTPRTF, CHGPRTF, and OVRPRTF commands. As a result, the system might not print the double-byte data as you expected and the following occurs:

- If a record to be printed exceeds the page width, the printer **wraps** the data (continues printing the record on the next line). Because the system is not aware that the data is wrapped, the system does not skip lines and start new pages properly. A new page might start in the middle of a record.
- The printer does not split double-byte characters when there is not enough room at the end of a line and a field of double-byte data is continued on a second printed line, even if you specified the CHRSIZ keyword. Instead, the system leaves a blank space on the first line where the character would have been printed and continues printing the complete character on the next line.

Spanned Pages (SCS DBCS Printers Only)

If data from a printed DBCS field spans to a second page, the system inserts a shift-in character at the beginning of each printed page of double-byte data, shifting the data out of DBCS mode. The printed data that follows does not make sense unless the data on the second page begins with a shift-out character.

To avoid this problem, break double-byte data fields that might span pages into several smaller fields.

Using the Print Key

If you want to print a display containing double-byte data by pressing the Print key, make sure that the associated display file or printer file is a DBCS file. If neither is a DBCS file, the display will not print properly.

One way to make sure that either the display or printer file is a DBCS file is to override the file using the OVRDSPF or the OVRPRTF command. For example, to override the system-supplied default printer file (the printer file used to print displays that are printed by pressing the Print key), enter:

```
OVRPRTF FILE(QSYSPRT) IGCDTA(*YES)
```

Notes:

1. If you do not plan to use double-byte data, do not change the printer file QSYSPRT to a DBCS file with a CHGPRTF command. This printer file is used to print a variety of system data, including alphanumeric data. A system performance degradation results if QSYSPRT is a DBCS file and it is processing only alphanumeric data.
2. If the Print key is used to print the image of a display containing DBCS-graphic data type fields, the system inserts shift-out and shift-in (SO/SI) characters around the graphic data. Depending on the IGCSOSI printer file value, the SO/SI characters could print as blanks, causing the printed alignment to be different from what was displayed.

Refer to the File Managementtopic in the iSeries Information Center for more information on overrides.

5553 Printer End-of-Forms

If you send the ignore (I) reply to the end-of-forms message that you receive when using continuous forms on the 5553 Printer, and if the printer has already printed within 2-1/2 inches of the bottom of the page, the system might not start printing subsequent pages where expected.

To avoid this problem, do the following when you receive the end-of-forms message:

1. Remove the current form from the tractor feed.
2. Insert new forms.
3. Align the first form to the first line.
4. Press the CANCEL button on the printer.
5. Press the SELECT button on the printer.
6. Respond to the end-of-forms message:
   a. For spooled files, specify the page on which you want to continue printing when you enter a response to the message. Determine which page to continue printing as follows:
      1) If no data was printed on the last 2-1/2 inches of the last form, enter the number of the next page to be printed.
      2) If data was printed on the last 2-1/2 inches of the last form, enter the number of the last page printed. Reprinting the page ensures that all of the data is printed.

      Use the Work with Writer (WRKWTR) command to find out approximately which page was last printed. The WRKWTR command displays the number of pages that the writer has currently printed.
   b. For direct printer output, enter RETRY to reprint the last page printed. This ensures that all of the data is printed.

Effects of Printing Double-Byte Data on Alphanumeric Printers

Printing DBCS output on an alphanumeric printer can result in degradation of system performance.

In addition, the following occurs using printer files that are indicated to be DBCS-capable through DDS or with the IGCDTA parameter:

- For direct printer output, the system prints the file and sends a diagnostic message describing the situation to your program message queue.

  Instead of printing double-byte data, the system prints double-byte characters as underscores (__) and prints shift-control characters as blanks (   ). Although the system does not print the individual double-byte characters, they are correctly stored in the system.

- For spooled printer output, the system sends an inquiry message to the message queue named on the Start Printer Writer (STRPRTWTR) command. This message lets you do the following:

  – Continue printing. When you continue printing, the system prints the file but does not print double-byte characters within it. Instead, the system prints double-byte characters as underscores (__) and prints shift-control characters as blanks (   ). Although the system does not print the individual double-byte characters, they are correctly stored in the system.

  – Hold the spooled file so that you can transfer it to an output queue used only for DBCS output. See Chapter 3, "Spool support" for instructions on transferring a spooled file.

  – Cancel printing altogether.

Spool Support

Create separate output queues for double-byte and alphanumeric data. This may improve throughput (the rate at which the system processes work) because the system can process alphanumeric data more quickly than it can process double-byte data.

See theWork Management topic in the iSeries Information Center for detailed information about creating output queues.

Applying Overrides in Printing

When starting a job, consider adding the OVRPRTF command to the initial program of the job:

```
OVRPRTF FILE(QSYSPRT) IGCDTA(*YES)
```

Override the printer file (QSYSPRT) to make it capable of printing double-byte data and to ensure that DBCS output printed as the result of pressing the Print key is printed properly. Refer to theFile Management topic in the iSeries Information Center for more information on overrides.

3130 Printer Resident Font Support

When printing to a 3130 printer (Release 2 of printer microcode required), you may specify to use the double byte fonts resident in the 3130 printer.

The QPQCHGCF program provides a way for you to indicate whether a particular section of a double byte coded font is resident in the printer or should be downloaded.

This section provides the following information:
- How to use the QPQCHGCF program
- Examples of using the QPQCHGCF program .
- Restrictions on using the QPQCHGCF program
- List of IBM supplied coded fonts whose font character sets are resident in the 3130 printer.
- QPQCHGCF instructions for marking coded fonts.

How to use the QPQCHGCF program.

Parameters:

1	Coded font name	Input	Char(8)
2	Coded font library name	Input	Char(10)
3	Font character set name	Input	Char(10)
4	Resident font indicator	Input	Char(4)

The QPQCHGCF program provides a way for you to indicate a particular section of a double byte coded font is

- Resident in the printer and should not be downloaded

- Is not resident in the printer or has been changed (different version than one in the printer) and needs to be downloaded by PSF/400 to the printer.

QPQCHGCF has the following parameters:

Coded-font-name:
Specifies the name of the coded font to be marked. This is a 8 character input parameter.

Coded-font-library-name
Specifies the name of the library containing the coded font. This is a 10 character input parameter.

You can use the following special values for the library name:

*LIBL This indicates that the job's current library list will be used to search for the coded font.

Font-character-set-name:
Specifies the name of the font character set to mark within the coded font. This is a 8 character input parameter.

The font character set name can be specified with the following special value:

*ALL This indicates that all the font character set/code page pairs in the coded font are to be marked.

The font character set name may be a generic name. A generic name is a character string of one or more characters followed by an asterisk (*); for example, C0S0*. The asterisk substitutes for any valid characters. A generic name specifies all font character sets with names that begin with the generic prefix. If an asterisk is not included in the name, the system assumes it to be the complete font character set name. To change all the font character sets in a certain range, for example C0G16F70 - C0G16F7F, you should specify C0G16F7* for the font character set name. Specifying C0G16F* would be the same as *ALL and change all the font character sets in the coded font (assuming all font character set names started with C0G16F0).

Resident-font-indicator:
Specifies whether the font character set is resident in the printer or if it is not resident in the printer and needs to be downloaded by the system.

*NO: The font character set is not resident in the printer and needs to be downloaded by the system to the printer. Also, the font character set may be resident in the printer, but has been modified. In that case, *NO should be specified.

:PK *YES: The font character set is resident in the printer and does not need to be downloaded by the system to the printer.

Note:

All IBM supplied coded fonts are shipped with the resident font indicator turned off. That means the entire font will be downloaded unless the QPQCHGCF program is run to mark the IBM supplied sections as resident.

The font character set and code page pair are treated the same for marking them resident or need to be downloaded. If the font character set has changed, then the corresponding code page will also be marked as needing to downloaded. If the code page has been changed, then the corresponding font character will also be marked as needing to downloaded.

Examples of using QPQCHGCF

Example 1: Example below marks all the font character set/code page pairs as resident in the printer for coded font X0G16F in library QFNT61. No user defined sections will be downloaded.

```
CALL QPQCHGCF  (X0G16F QFNT61 *ALL *YES)
```

Example 2: Example below marks all the font character set/code page pairs in sections 41 - 4F as resident and then marks sections 50 - 55 as resident.

```
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F4* *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F50 *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F51 *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F52 *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F53 *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F54 *YES)
CALL QPQCHGCF  (X0G16B QFNT61 C0G16F55 *YES)
```

Example 3: Example below marks all the font character set/code page pairs in sections 41 - 4F as resident; then marks section 48 to be downloaded. Sections 50 - 5F are marked as resident and sections 60 - 68 are marked as resident.

```
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F4* *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F48 *NO)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F5* *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F60 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F61 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F62 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F63 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F64 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F65 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F66 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F67 *YES)
CALL QPQCHGCF  (X0G16F QFNT61 C0G16F68 *YES)
```

Restrictions on using the QPQCHGCF program

- When marking fonts, the print writer must be stopped and re-started If fonts are marked while the writer is active, unpredictable results will occur.
  - End PSF/400 (ENDWTR), if it is active.
  - Use QPQCHGCF to mark the coded font.
  - Start PSF/400 (STRPRTWTR).
- If a section of a font is modified, the modified section must contain all rotations that the original font contained. For example if the coded font X0M16B is resident in the printer in rotations 0, 90,180 and 270, and section 46 is modified, then the modified section 46 must also contain rotations 0, 90, 180, and 270.
- If IBM supplied coded fonts are modified, you should not remove sections from the coded font as supplied. This could result in incorrect results when the modified font is referenced in a job being printed on device that does not support resident double byte raster fonts.
- PSF/400 does not support referencing a double byte resident raster font by its registered font ID. That is, you should not specify a double byte font on the FONT parameter of the printer file, on the FONT DDS keyword, or with any other application that allows you to specify font with its registered font ID.

Coded fonts whose font character sets are resident in the 3130

The following is a list of the DBCS fonts that are resident in the 3130 printer.

```
Japanese (In QFNT61 library)
FONT          RESIDENT FONT
FONT          SIZE    CODED FONT  CHARACTER SET   FONTID WIDTH
------------------------------------------------------------
Mincho        16x16   X0M16B/F    C0M16FXX        53559  096
Mincho        24x24   X0M24B/F    C0M24FXX        53559  140
Mincho        20x24   X0Z24B/F    C0Z24FXX        53559  144
Mincho        26x26   X0M26B/F    C0M26FXX        53559  156
Mincho        32x32   X0M32B/F    C0M32FXX        53559  180
Mincho        36x36   X0M36B/F    C0M36FXX        53559  216
Mincho        40x40   X0M40B/F    C0M40FXX        53559  240
Mincho        44x44   X0M44B/F    C0M44FXX        53559  264
Mincho        48x48   X0M48B/F    C0M48FXX        53559  288
Mincho        52x52   X0M52B/F    C0M52FXX        53559  312
Mincho        64x64   X0M64B/F    C0M64FXX        53559  384
Gothic        16x16   X0G16B/F    C0G16FXX        53815  100
Gothic        20x24   X0G20B/F    C0G20FXX        53813  144
Gothic        24x30   X0G24B/F    C0G24FXX        53813  140
Gothic        32x32   X0G32B/F    C0G32FXX        53815  192
Gothic        36x36   X0G36B/F    C0G36FXX        53815  216
Gothic        48x48   X0G48B/F    C0G48FXX        53815  288
Gothic        64x44   X0G64B/F    C0G64FXX        53815  384
R-Gothic      36x36   X0R36B/F    C0R36FXX        54071  216
R-Gothic      40x40   X0R40B/F    C0R40FXX        54071  240
R-Gothic      48x48   X0R48B/F    C0R48FXX        54071  288
R-Gothic      64x64   X0R64B/F    C0R64FXX        54071  384


Korean (In QFNT62 library)
FONT          RESIDENT FONT
FONT          SIZE    CODED FONT  CHARACTER SET   FONTID WIDTH
------------------------------------------------------------
Mincho        24x24   X0M24K/L    C0HB00XX        53559  144
Mincho        32x32   X0M32K/L    C0HD00XX        53559  192
Mincho        36x36   X0M36K/L    C0HE00XX        53559  216
Mincho        40x40   X0M40K/L    C0HF00XX        53559  240
Mincho        48x48   X0M48K/L    C0HG00XX        53559  288
Mincho        64x64   X0M64K/L    C0HH00XX        53559  384
Gothic        16x16   X0G16K/L    C0HA00XX        53815  096
Gothic        24x30   X0G24K/L    C0HC00XX        53813  180


Traditional Chinese (In QFNT63 library)
FONT          RESIDENT FONT
FONT          SIZE    CODED FONT  CHARACTER SET   FONTID WIDTH
------------------------------------------------------------
Ming          24x24   X0M24T      C0TB00XX        54583  144
Ming          32x32   X0M32T      C0TC00XX        54583  192
Ming          40x40   X0M40T      C0TD00XX        54583  240
Gothic        16x16   X0G16T      C0TA00XX        53815  096

Simplified Chinese (In QFNT64 library)
FONT          RESIDENT FONT
FONT          SIZE    CODED FONT  CHARACTER SET   FONTID WIDTH
------------------------------------------------------------
Song          26x26   X0S26P      C0S26PXX        54327  144
Song          32x32   X0S32P      C0S32PXX        54327  192
Song          40x40   X0S40P      C0S40PXX        54327  240
Gothic        16x16   X0G16P      C0G16PXX        53815  096


Thai (In QFNT65 library)
FONT          RESIDENT FONT
FONT          SIZE    CODED FONT  CHARACTER SET   FONTID WIDTH
------------------------------------------------------------
Official      24x40   X0O40F      C0O40FXX        57655  240
Official      24x60   X0O60F      C0O60FXX        57655  360
Italics       24x60   X0I60F      C0I60FXX        58039  360
```

QPQCHGCF instructions for marking coded fonts.

Included with the QPQCHGCF program is file QCDEFNT in library QGPL This file contains five members: QFNT61, QFNT62, QFNT63, QFNT64, and QFNT65. Each member contains CL statements for marking the IBM supplied sections of the fonts shipped in libraries QFNT61 - QFNT65.

To run a batch job to mark IBM supplied sections as resident, you must edit the appropriate member in QCDEFNT.

- Edit the JOBD parameter on the //BCHJOB statement. This JOBD should have enough authority to change the coded fonts in the QFNT6X libraries.
- Edit this source file if you have changed any IBM supplied sections and want these sections to be downloaded.

After the source file has been edited, you can run the batch job to mark the specified fonts.

This is done with the STRDBRDR command

```
STRDBRDR  FILE(QGPL/QCDEFNT)
MBR(QFNT6X)
```

where member-name is QFNT61, QFNT62, QFNT63, QFNT64, OR QFNT65.

In order that the marked coded fonts take effect, you must perform the following:

- End PSF/400 (ENDWTR), if it is active.
- Run batch program to mark coded fonts.
- Start PSF/400 (STRPRTWTR).

QCDEFNT in QUSRSYS contains the following five members:

```
QFNT61  - CL statements to mark Japanese coded fonts
QFNT62  - CL statements to mark Korean coded fonts
QFNT63  - CL statements to mark Traditional Chinese coded fonts
QFNT64  - CL statements to mark Simplified Chinese coded fonts
QFNT65  - CL statements to mark Thai coded fonts
```

Appendix G. Feedback Area Layouts

This chapter contains general-use programming interface and associated guidance information

Tables in this section describe the open and I/O feedback areas associated with any opened printer file. The following information is presented for each item in these feedback areas:

- Offset, which is the number of bytes from the start of the feedback area to the location of each item
- Data type
- Length, which is given in number of bytes
- Contents, which is the description of the item and the valid values for it

The support provided by the high-level language you are using determines how to access this information and how the data types are represented. See your high-level language manual for more information.

Note: The tables in this appendix are relevant only to device file type of printer. For a complete description of all file types (printer, diskette, displays, tape, database, ICF, and inline) see the File Managementtopic in the iSeries Information Center.

Open Feedback Area for Printer

The **open data path** (ODP) contains information about the merged file attributes and information returned by input or output operations. The ODP only exists while the file is open.

The **open feedback area** is the part of the open data path (ODP) that contains general information about the file after it has been opened. It also contains file-specific information, depending on the file type, plus information about each device defined for the file. This information is set during open processing and may be updated as other operations are performed.

Table 52. Open Feedback Area

Offset	Data Type	Length in Bytes	Contents	
0	Character	2	Open data path (ODP) type:	
			DS	Printer file not being spooled, display, tape, ICF, save, or diskette file not being spooled.
			SP	Printer or diskette file being spooled or inline data file.
2	Character	10	Name of the file being opened. If the ODP type is DS, this is the name of the device file or save file. If the ODP type is SP, this is the name of the device file or the inline data file.	
12	Character	10	Name of the library containing the file.	
22	Character	10	Name of the spooled file. The name of a database file containing the spooled input or output records.	

Table 52. Open Feedback Area (continued)

Offset	Data Type	Length in Bytes	Contents
32	Character	10	Name of the library in which the spooled file is located.
42	Binary	2	Spooled file number.
44	Binary	2	Maximum record length.
46	Character	2	Reserved.
48	Character	10	Member name: • If ODP type SP, the member name in the file named at offset 22.
58	Binary	4	Reserved.
62	Binary	4	Reserved.
66	Binary	2	File type: **1** Display **2** Printer **4** Diskette **5** Tape **9** Save **10** DDM **11** ICF **20** Inline data **21** Database
68	Character	3	Reserved.
71	Binary	2	Number of lines on a printed page.
73	Binary	2	Number of positions on a printed line.
75	Binary	4	Not applicable to printer.
79	Binary	4	Spooled file number.
83	Character	10	Reserved.
93	Character	10	Reserved.
103	Binary	2	Not applicable to printer.
105	Binary	2	Maximum number of records that can be read or written in a block when using blocked record I/O.
107	Binary	2	Overflow line number.
109	Binary	2	Blocked record I/O record increment. Number of bytes that must be added to the start of each record in a block to address the next record in the block.
111	Binary	4	Reserved.

115	Character	1	Miscellaneous flags.
			
			 Bit 1: Reserved.
			
			 Bit 2: File shareable
			
			      0 File was not opened shareable.
			
			      1 File was opened shareable (SHARE(*YES)).
			
			 Bits 3-5:
			 Not applicable to printer.

Table 52. Open Feedback Area (continued)

Offset	Data Type	Length in Bytes	Contents
			Bit 6: Field-level descriptions
			0 File does not contain field-level descriptions.
			1 File contains field-level descriptions.
			Bit 7: DBCS-capable file
			0 File is not DBCS-capable.
			1 File is DBCS-capable.
			Bit 8: Not applicable to printer.
116	Character	10	Not applicable to printer.
126	Binary	2	File open count. If the file has not been opened shareable, this field contains a 1. If the file has been opened shareable, this field contains the number of programs currently attached to this file.
128	Binary	2	Reserved.
130	Binary	2	Not applicable to printer.
132	Character	1	Miscellaneous flags.
			Bits 1-4: Not applicable to printer.
			Bit 5: Separate indicator area
			0 Indicators are in the I/O buffer of the program.
			1 Indicators are not in the I/O buffer of the program. The DDS keyword, INDARA, was used when the file was created.
			Bit 6: User buffers
			0 System creates I/O buffers for the program.
			1 User program supplies I/O buffers.
			Bits 7-8: Reserved.
133	Character	2	Open identifier. The value is unique for a full (not shared) open operation of a file. It allows you to match this file to an entry on the associated data queue.

Table 52. Open Feedback Area (continued)

Offset	Data Type	Length in Bytes	Contents
135	Binary	2	The field value is the maximum record format length, including both data and file-specific information such as: first-character forms control, option indicators, response indicators, source sequence numbers, and program-to-system data. If the value is zero, then use the field at offset 44.
137	Character	9	Reserved.
146	Binary	2	Number of devices defined for this ODP. For printers this always has a value of 1.
148	Character		Device name definition list. See "Device Definition List" for a description of this array.

Device Definition List

The device definition list part of the open feedback area is an array structure. Each entry in the array contains information about each device or communications session attached to the file. The number of entries in this array is determined by the number at offset 146 of the open feedback area. The device definition list begins at offset 148 of the open feedback area. The offsets shown for it are from the start of the device definition list rather than the start of the open feedback area.

Table 53. Device Definition List

Offset	Data Type	Length in Bytes	Contents
0	Character	10	Program device name. For printer or diskette files being spooled, the value is *N. For database files, the value is DATABASE. For save files, the value is *NONE. For ICF files, the value is the name of the program device from the ADDICFDEVE or OVRICFDEVE command. For all other files, the value is the name of the device description.
10	Character	50	Reserved.
60	Character	10	Device description name. For printer or diskette files being spooled, the value is *N. For save files, the value is *NONE. For all other files, the value is the name of the device description.
70	Character	1	Device class.
			hex 01 Display
			hex 02 Printer
			hex 04 Diskette
			hex 05 Tape
			hex 09 Save
			hex 0B ICF

Table 53. Device Definition List (continued)

Offset	Data Type	Length in Bytes	Contents
71	Character	1	Device type.
			hex 02 5256 Printer
			hex 0C 5224/5225 printers
			hex 0F 5219 Printer
			hex 10 5583 Printer (DBCS)
			hex 11 (DBCS) 5553 Printer
			hex 14 3270 Printer
			hex 21 4234 (SCS) Printer
			hex 22 3812 (SCS) Printer
			hex 23 4214 Printer
			hex 24 4224 (IPDS) Printer
			hex 25 4245 Printer
			hex 29 5262 Printer
			hex 30 3812 (IPDS) Printer
			hex 31 4234 (IPDS) Printer
			hex 32 IPDS printer, model unknown
			hex 55 6252 (SCS) Printer
			hex 57 4230 (IPDS) Printer
			hex 63 3935 (IPDS) Printer
72	Binary	2	Not applicable to printer.
74	Binary	2	Not applicable to printer.
76	Character	2	Not applicable to printer.
78	Character	1	Not applicable to printer.
79	Character	1	Not applicable to printer.
80	Character	50	Reserved.

I/O Feedback Area

The results of I/O operations are communicated to the program using OS/400 messages and I/O feedback information. The I/O feedback area is updated for every I/O operation unless your program is using blocked record I/O. In that case, the feedback area is updated only when a block of records written. Some of the information reflects the last record in the block. Other information, such as the count of I/O operations, reflects the number of operations on blocks of records and not the number of records. See your high-level language manual to determine if your program uses blocked record I/O.

The I/O feedback area consists of two parts: a common area and a file-dependent area. The file-dependent area varies by the file type. This guide discusses device file type of printers only.

Common I/O Feedback Area

Table 54. Common I/O Feedback Area

Offset	Data Type	Length in Bytes	Contents
0	Binary	2	Offset to file-dependent feedback area.
2	Binary	4	Write operation count. Updated only when a write operation completes successfully. For blocked record I/O operations, this count is the number of blocks, not the number of records.
6	Binary	4	Read operation count. Not applicable to printers.
10	Binary	4	Write-read operation count. Not applicable to printers.
14	Binary	4	Other operation count. Number of successful operations other than write, read, or write-read. Updated only when the operation completes successfully. This count includes force-end-of-data.
18	Character	1	Reserved.
19	Character	1	Current operation. **hex 05** Write or write block **hex 09** Force-end-of-data
20	Character	10	Name of the record format just processed, which is either: • Specified on the I/O request, or • Determined by default or format selection processing

Table 54. Common I/O Feedback Area (continued)

Offset	Data Type	Length in Bytes	Contents
30	Character	2	Device class:

Byte 1:

hex 00   Database

hex 01   Display

hex 02   Printer

hex 04   Diskette

hex 05   Tape

hex 09   Save

hex 0B   ICF

Byte 2 (if byte 1 is 02 for printer):

hex 02   5256 Printer

hex 0C   5224/5225 printers

hex 0F   5219 Printer

hex 10   5583 Printer (DBCS)

hex 11   5553 Printer (DBCS)

hex 14   3270 Printer

hex 21   4234 (SCS) Printer

hex 22   3812 (SCS) Printer

hex 23   4214 Printer

hex 24   4224 (IPDS) Printer

hex 25   4245 Printer

hex 29   5262 Printer

hex 30   3812 (IPDS) Printer

hex 31   4234 (IPDS) Printer

hex 32   IPDS printer, model unknown

hex 55   6252 (SCS) Printer

hex 57   4230 (IPDS) Printer

hex 63   3935 (IPDS) Printer

Offset	Data Type	Length in Bytes	Contents
32	Character	10	Device name. The name of the device for which the operation just completed. Supplied only for printer, display, tape, diskette, and ICF files. For printer files being spooled, the value is *N. For printer files not being spooled, the value is the device description name.
42	Binary	4	Not applicable to printer.
46	Character	80	Reserved.
126	Binary	2	Not applicable to printer.
128	Binary	2	For printers, the field value is the record format length, including first-character forms control, option indicators, source sequence numbers, and program-to-system data. If the value is zero, use the field at offset 42.

Table 54. Common I/O Feedback Area (continued)

Offset	Data Type	Length in Bytes	Contents
130	Character	2	Reserved.
132	Binary	4	Not applicable to printer.
136	Character	8	Reserved.

I/O Feedback Area for Printer Files

Table 55. I/O Feedback Area for Printer Files

Offset	Data Type	Length in Bytes	Contents
0	Binary	2	Current line number in a page.
2	Binary	4	Current page count.
6	Character	1	Miscellaneous flags
			Bit 1: Spooled file deleted.
			Bit 2–8: Reserved.
			.
7	Character	27	Reserved.
34	Character	2	Major return code.
			00 Operation completed successfully
			80 Permanent system or file error
			81 Permanent device error
			82 Open operation failed
			83 Recoverable device error occurred
36	Character	2	Minor return code.

Appendix H. Using DDS with High-Level Languages (HLL)

This appendix contains examples of DDS used with COBOL and RPG. Two different DDS are used with each programming language.

One DDS uses row/column as the positioning method for the data being acted on. The other DDS uses absolute positioning as its positioning method. When absolute positioning is used in DDS, all other objects (page segments, boxes, rotated text) must use absolute positioning.

Data Description Specifications (DDS)

Figure 10 and Figure 11 show the DDS source for:
- Row/column method of positioning
- Absolute method of positioning

DDS Coding Example Using the Row/column Method of Positioning

The row/column method of positioning means specifying where the data starts printing (how many rows down and how many columns in).

The following figure shows DDS coding using the row/column method of positioning.

```
000100911101              R LABEL
000200911101                NAME       25A  0  8 10
000300911101                ADDR1      25A  0  9 10
000400911101                CITY       15A  0 10 10
000500911101                STATE       2A  0 10 27
000600911101                ZIPCD       5S 00 10 30BARCODE(POSTNET *HRITOP)
000700911101                            13  5'Made in the USA'
000800911101                               TXTRTT(270)
000900911101                               FONT(5687 (*POINTSIZE 6))
```

Figure 10. DDS Source Using Row/Column Method of Positioning

For example, at row 10 column 30, a bar code is printed.

DDS Coding Example Using the Absolute Method of Positioning

Absolute positioning means being able to start printing at any point on a piece of paper by specifying that point.

Use of absolute positioning requires the Advanced Function Printing data stream (AFPDS). This is obtained on OS/400 by specifying *AFPDS on the device type (DEVTYPE) parameter of the printer file.

Inches or centimeters are the measurement methods available. You choose the measurement method by specifying *INCH or *CENT on the unit of measure (UOM) parameter of the printer file.

The following figure shows DDS coding using the absolute method of positioning.

```
000100911101          R LABEL                    PAGSEG(LOGO 1.5 1)
000200911101                                      BOX(1 1 5 4 *MEDIUM)
000300911101                                      BOX(1.5 4 2 4.5 *NARROW)
000400911101            NAME       25A  O         POSITION(1.3 1.6)
000500911101            ADDR1      25A  O         POSITION(1.5 1.6)
000600911101            CITY       15A  O         POSITION(1.7 1.6)
000700911101            STATE       2A  O         POSITION(1.7 2.7)
000800911101            ZIPCD       5S 00         POSITION(1.7 3)
000900911101                                      BARCODE(POSTNET *HRITOP)
f01000911101            TEXT       20A  O         TXTRTT(270)
000800911101                                      POSITION(1.9 .25)
001100911101                                      FONT(5687 (*POINTSIZE 6))
```

Figure 11. DDS Source Using Absolute Positioning

In this example, a page segment called LOGO is specified to start printing at 1.5 units down and 1 unit in. The TEXT record (Made in the USA) is supplied by the application program. The UOM parameter value (*INCH or *CM) of the printer file determines which unit of measurement is used.

COBOL and RPG Source Code

The following figures that contain the COBOL and RPG source code that produces (along with the previously discussed DDS source) the examples in the next section.

The following figure contains the COBOL source.

```
STMT SEQNBR -A 1 B..+....2....+....3....+....4....+....5....+....6....+....7..
IDENTFCN  S  COPYNAME                                                    CHG DATE
  1  000100 IDENTIFICATION DIVISION.                                     10/20/91
  2  000200 PROGRAM-ID.   CBLLBL.                                        10/20/91
  3  000300 ENVIRONMENT DIVISION.                                        10/20/91
  4  000400 INPUT-OUTPUT SECTION.                                        10/20/91
  5  000500 FILE-CONTROL.                                                10/20/91
  6  000600     SELECT PRINTER-FILE                                      10/20/91
  7  000700        ASSIGN TO FORMATFILE-LABELS.                          10/23/91
  8  000800     SELECT VENDOR-FILE                                       10/20/91
  9  000900        ASSIGN TO DATABASE-VENDORS.                           10/20/91
 10  001000 DATA DIVISION.                                               10/20/91
 11  001100 FILE SECTION.                                                10/20/91
 12  001200 FD  PRINTER-FILE                                             10/20/91
 13  001300     DATA RECORD IS PRINT-REC.                                10/23/91
 14  001400 01  PRINT-REC.                                               10/23/91
 15  001500     COPY DDS-ALL-FORMATS-O OF LABELS.                        10/23/91
 16 +000001     05  LABELS-RECORD PIC X(72).                            <-ALL-FMTS
    +000002* OUTPUT FORMAT;LABEL    FROM FILE LABELS    OF LIBRARY SGAFP <-ALL-FMTS
    +000003*                                                            <-ALL-FMTS
 17 +000004     05  LABEL-O       REDEFINES LABELS-RECORD.              <-ALL-FMTS
 18 +000005         06 NAME                 PIC X(25).                  <-ALL-FMTS
 19 +000006         06 ADDR1                PIC X(25).                  <-ALL-FMTS
 20 +000007         06 CITY                 PIC X(15).                  <-ALL-FMTS
 21 +000008         06 STATE                PIC X(2).                   <-ALL-FMTS
 22 +000009         06 ZIPCD                PIC S9(5).                  <-ALL-FMTS
 23  001600 FD  VENDOR-FILE                                              10/23/91
 24  001700     DATA RECORD IS VENDOR-REC.                               10/23/91
 25  001800 01  VENDOR-REC.                                              10/23/91
 26  001900     COPY DDS-ALL-FORMATS-I OF VENDORS.                       10/23/91
 27 +000001     05  VENDORS-RECORD PIC X(82).                           <-ALL-FMTS
    +000002*   I-O FORMAT;VNDMSTR  FROM FILE VENDORS  OF LIBRARY SGAFP  <-ALL-FMTS
    +000003*                            VENDMAST DB FORMAT              <-ALL-FMTS
 28 +000004     05  VNDMSTR       REDEFINES VENDORS-RECORD.             <-ALL-FMTS
 29 +000005         06 VNDNBR              PIC S9(5)       COMP-3.      <-ALL-FMTS
    +000006*                       VENDOR NUMBER                        <-ALL-FMTS
 30 +000007         06 NAME                PIC X(25).                   <-ALL-FMTS
    +000008*                       NAME                                 <-ALL-FMTS
 31 +000009         06 ADDR1               PIC X(25).                   <-ALL-FMTS
    +000010*                       ADDRESS LINE 1                       <-ALL-FMTS
 32 +000011         06 CITY                PIC X(15).                   <-ALL-FMTS
    +000012*                       CITY                                 <-ALL-FMTS
 33 +000013         06 STATE               PIC X(2).                    <-ALL-FMTS
    +000014*                       STATE                                <-ALL-FMTS
 34 +000015         06 ZIPCD               PIC S9(5)       COMP-3.      <-ALL-FMTS
    +000016*                       ZIP CODE                             <-ALL-FMTS
 35 +000017         06 VNDCLS              PIC S9(2)       COMP-3.      <-ALL-FMTS
    +000018*                       VENDOR CLASS                         <-ALL-FMTS
 36 +000019         06 VNDSTS              PIC X(1).                    <-ALL-FMTS
    +000020*                       A=ACTIVE, D=DELETE, S=SUSPEND        <-ALL-FMTS
 37 +000021         06 BALOWE              PIC S9(7)V9(2)  COMP-3.      <-ALL-FMTS
    +000022*                       BALANCE OWED                         <-ALL-FMTS
 38 +000023         06 SRVRTG              PIC X(1).                    <-ALL-FMTS
    +000024*                       G=GOOD, A=AVERAGE, B=BAD, P=PREFERRED<-ALL-FMTS
 39  002000 WORKING-STORAGE SECTION.                                     10/20/91
 40  002100 77  EOF-FLAG            PIC X.                               10/23/91
 41  002200     88  NOT-END-OF-FILE           VALUE " ".                10/23/91
```

Figure 12. COBOL Source for DDS Example (panel 1)

```
5738CB1 V2R1M0  910524
AS/400 COBOL Source  SGAFP/CBLLBL  RCHASA12  10/24/91 10:18:16  Page  3
STMT SEQNBR -A 1 B..+....2....+....3....+....4....+....5....+....6....+....7..
IDENTFCN  S  COPYNAME                                                CHG DATE
  42 002300     88 END-OF-FILE                   VALUE "1".          10/23/91
     002400                                                          10/23/91
  43 002500 PROCEDURE DIVISION.                                      10/23/91
     002600 MAIN-PARA.                                               10/23/91
  44 002700    OPEN INPUT VENDOR-FILE                                10/23/91
     002800         OUTPUT PRINTER-FILE.                             10/23/91
  45 002900    PERFORM PRINT-LABELS UNTIL END-OF-FILE.               10/23/91
  46 003000    CLOSE VENDOR-FILE, PRINTER-FILE.                      10/23/91
     003100    GOBACK.                                               10/23/91
     003200                                                          10/23/91
  47 003300 PRINT-LABELS.                                            10/23/91
  48 003400    READ VENDOR-FILE                                      10/23/91
  49 003500         AT END SET END-OF-FILE TO TRUE.                  10/23/91
  50 003600    IF NOT-END-OF-FILE                                    10/23/91
  51 003700       MOVE CORRESPONDING VNDMSTR TO LABEL-O              10/23/91
        *       ** CORRESPONDING items for statement 51:
        *       **    NAME
        *       **    ADDR1
        *       **    CITY
        *       **    STATE
        *       **    ZIPCD
        *       ** End of CORRESPONDING items for statement 51
  52 003800       WRITE PRINT-REC FORMAT IS "LABEL".                 10/23/91
* * * * *  E N D   O F   S O U R C E   * * * * *
```

Figure 13. COBOL Source for DDS Example (Panel 2)

The following figure contains the RPG source.

```
SEQUENCE        IND    DO    LAST      PAGE    PROGRAM
NUMBER     *...1....+....2....+....3....+....4....+....5....+....6....+....7...*
  USE    NUM   UPDATE     LINE   ID
                          S o u r c e   L i s t i n g
         H                                                            *****
    100  FVENDORS IF  E          K        DISK                    10/24/91
           RECORD FORMAT(S): LIBRARY SGAFP FILE VENDORS.
                    EXTERNAL FORMAT VNDMSTR RPG NAME VNDMSTR
    200  FLABELS  O   E                    PRINTER                08/25/91
           RECORD FORMAT(S): LIBRARY SGAFP FILE LABELS.
                    EXTERNAL FORMAT LABEL RPG NAME LABEL
A000000   INPUT  FIELDS FOR RECORD VNDMSTR FILE VENDORS FORMAT VNDMSTR.
A000000       VENDMAST DB FORMAT
A000001                              P   1   30VNDNBR   VENDOR NUMBER
A000002                                  4  28 NAME      NAME
A000003                                 29  53 ADDR1     ADDRESS LINE 1
A000004                                 54  68 CITY      CITY
A000005                                 69  70 STATE     STATE
A000006                              P  71  730ZIPCD     ZIP CODE
A000007                              P  74  750VNDCLS    VENDOR CLASS
A000008                                 76  76 VNDSTS    A=ACTIVE, D=DELETE, S=SUSPEND
A000009                              P  77  812BALOWE    BALANCE OWED
A000010                                 82  82 SRVRTG    G=GOOD, A=AVERAGE, B=BAD, P=PREFERRED
    300  C                READ VENDORS                  50    3         10/24/91
    400  C        *IN50   DOWNE*ON                              B001    08/25/91
    500  C                WRITELABEL                            001     08/25/91
    600  C                READ VENDORS                  50    3  001    10/24/91
    700  C                ENDDO                                 E001    08/25/91
    800  C                SETON                         LR    1         08/25/91
B000000   OUTPUT FIELDS FOR RECORD LABEL FILE LABELS FORMAT LABEL.
B000001                          NAME      25  CHAR   25
B000002                          ADDR1     50  CHAR   25
B000003                          CITY      65  CHAR   15
B000004                          STATE     67  CHAR    2
B000005                          ZIPCD     72  ZONE   5,0
* * * * *  E N D   O F   S O U R C E   * * * * *
```

Figure 14. RPG Source for DDS Example

Example Output from the DDS, COBOL, and RPG Source

The following examples show the type of output you can get using DDS in conjunction with the absolute and row/column positioning methods.

Example 1: DDS and Row/Column Positioning

The following example shows the output achieved using the row/column positioning method.

World Wide Service Corp.
7539 N. Peachtree
Atlanta   GA

Made in the USA

RV2H336-1

Example 2: DDS and Absolute Positioning

This example highlights the additional function absolute positioning provides by using boxes (indicating where the stamp goes) and page segments (the pencil-type logo).
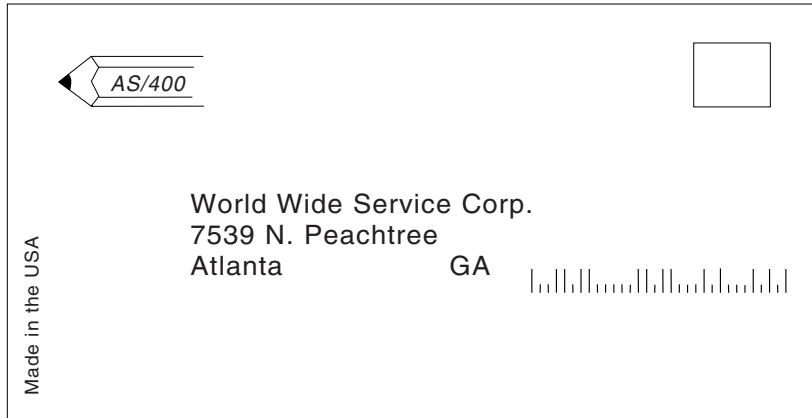
RV2H335-1

Appendix I. What Does a Font Look Like?

Did you ever wonder what a font looks like before using it in an application? This appendix contains instructions and source code that allow you to print an FGID, font character set, or coded font and see what it looks like. The source code provided is data description specifications (DDS), which can be used with the following high-level languages:

- C
- COBOL
- Pascal
- RPG

To print the FGID, font character set, or coded font you want to see, edit the DDS source and insert the correct identifier or name.

Notes:

1. These sample programs work only with printers configured as AFP(*YES).
2. See Appendix D, "Working with Fonts, Font Character Sets, Code Pages, CHRIDs, and Coded Fonts" on page 377 for the names of FGIDs, font character sets, and coded fonts.

Getting Started

Follow the instructions in the list below. If you need assistance with any of the CL commands, use the F4 (Prompt) key and then press the Help key on any of the parameters.

1. Create a library to contain the objects needed to print the font samples. In this example, the library is named FONTSAMPLE.

   ```
   CRTLIB FONTSAMPLE
   ```

2. Add FONTSAMPLE to your library list.

   ```
   ADDLIBLE FONTSAMPLE
   ```

3. Create a source physical file in FONTSAMPLE to contain your source code. In this example the source file is named SOURCE.

   ```
   CRTSRCPF FONTSAMPLE/SOURCE
   ```

4. Add a member named FONT to that physical file. This member is used for entering the source code for the printer file.

   ```
   ADDPFM FILE(FONTSAMPLE/SOURCE) MBR(FONT)
   ```

5. Edit the member FONT with the source entry utility (SEU).

   ```
   STRSEU SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(FONT) TYPE(PRTF)
   ```

   Type the DDS source code (shown in "DDS Source Code" on page 536) for the printer file. When you are done, press F3 to exit.

6. Create the printer file from the DDS source you just typed in.

   ```
   CRTPRTF FILE(FONTSAMPLE/FONT) SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(FONT)
   DEVTYPE(*AFPDS)
   ```

7. Choose the high-level language you will be using to produce printed output. The compiler for the high-level language you choose must be installed on your system. Samples are given for C, RPG, Pascal, and COBOL. Add a

member to the physical file SOURCE. Use one of the following names depending on which language you choose:

- CCODE for the C language
- COBOLCODE for the COBOL language
- PASCODE for the Pascal language
- RPGCODE for the RPG language

```
ADDPFM FILE(FONTSAMPLE/SOURCE) MBR(CCODE, COBOLCODE, PASCODE, or RPGCODE)
```

8. Edit the member (CCODE, COBOLCODE, PASCODE, or RPGCODE) with source entry utility (SEU). Type in the command that corresponds to the high-level language you are using. Then, type the program source (listed below these instructions) for <u>one</u> of the languages.

C: `STRSEU SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(CCODE) TYPE(C)`

RPG: `STRSEU SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(RPGCODE) TYPE(RPG)`

Pascal:
    `STRSEU SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(PASCODE) TYPE(PAS)`

COBOL:
    `STRSEU SRCFILE(FONTSAMPLE/SOURCE) SRCMBR(COBOLCODE) TYPE(CBL)`

When you are done, press F3 to exit.

9. Create the program, using the command below that corresponds to the language you chose.

C: `CRTCPGM PGM(FONTSAMPLE/CPGM) SRCFILE(FONTSAMPLE/SOURCE)`
    `SRCMBR(CCODE)`

RPG: `CRTRPGPGM PGM(FONTSAMPLE/RPGPGM) SRCFILE(FONTSAMPLE/SOURCE`
    `SRCMBR(RPGCODE)`

Pascal:
    `CRTPASPGM PGM(FONTSAMPLE/PASPGM) SRCFILE(FONTSAMPLE/SOURCE)`
    `SRCMBR(PASCODE) LANGLVL(*SYSTEM)`

COBOL:
    `CRTCBLPGM PGM(FONTSAMPLE/CBLPGM) SRCFILE(FONTSAMPLE/SOURCE)`
    `SRCMBR(COBOLCODE)`

10. Call the program that corresponds to the language you chose:

C: `Call CPGM`

RPG: `Call RPGPGM`

Pascal:
    `Call PASPGM`

COBOL:
    `Call CBLPGM`

11. The output from the program is sent to a spooled file named FONT. You cannot view the spooled file until it prints.

DDS Source Code

```
5738PW1 V2R2M0  920615                     SEU SOURCE LISTING
02/09/93 13:56:16                   PAGE    1
SOURCE FILE . . . . . . .   FONTSAMPLE/SOURCE
MEMBER  . . . . . . . . .   FONT
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
...+... 8 ...+... 9 ...+... 0
```

```
100             R REC1
200                                 6 10FONT(5)
300                                    'Rhetoric Orator FGID 5'
400                                 8 10FNTCHRSET(QFNT01/C0T055B0 +
500                                        QFNT01/T1V10037)
600                                    'Sonoran Serif 12 Pt +
700                                     Font Char Set C0T055B0'
800                                10 10CDEFNT(QFNTCPL/X0BIR1)
900                                    'Book Italic 10 Pt +
1000                                    CODED FONT X0BIR1'
1100      * Above is the source for DDS and the fontsample program
* * * * E N D  O F  S O U R C E  * * * *
```

C Source Code

```
5738PW1 V2R2M0  920615                  SEU SOURCE LISTING
SOURCE FILE . . . . . . .  FONTSAMPLE/SOURCE
MEMBER  . . . . . . . . .  CCODE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
...+... 8 ...+... 9 ...+... 0
100 #include <stdio.h>
200 #include <xxasio.h>
300 main()
400 {
500 FILE          *outfile;
600
700 outfile = fopen("font","wb type=record");
800 QXXFORMAT(outfile, "REC1      ");
900 fwrite("",0,0,outfile);
1000 }
* * * * E N D  O F  S O U R C E  * * * *
```

Pascal Source Code

```
5738PW1 V2R2M0  920615                  SEU SOURCE LISTING
SOURCE FILE . . . . . . .  FONTSAMPLE/SOURCE
MEMBER  . . . . . . . . .  PASCODE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
...+... 8 ...+... 9 ...+... 0
100 program print (input, output);
200 var
300   format:  BINDINGTYPE;
400   prtfile: FILE OF char;
500 begin
600   REWRITE(prtfile, 'file(font) COMMIT(*NO) FILETYPE(*PRTF)');
700   format.options := 'format(rec1)';
800   BIND(prtfile, format);
900   put(prtfile);
1000 end.
* * * * E N D  O F  S O U R C E  * * * *
```

RPG Source Code

```
5738PW1 V2R2M0  920615                  SEU SOURCE LISTING
SOURCE FILE . . . . . . .  FONTSAMPLE/SOURCE
MEMBER  . . . . . . . . .  RPGCODE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
...+... 8 ...+... 9 ...+... 0
100     FFONT   O  E                  PRINTER
200     C                    WRITEREC1
300     C                    SETON                  LR
* * * * E N D  O F  S O U R C E  * * * *
```

COBOL Source Code

```
5738PW1 V2R2M0  920615                      SEU SOURCE LISTING
SOURCE FILE . . . . . . .  FONTSAMPLE/SOURCE
MEMBER  . . . . . . . . .  COBOLCODE
SEQNBR*...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
...+... 8 ...+... 9 ...+... 0
100 100010 IDENTIFICATION DIVISION.
200
300 100020 PROGRAM-ID. PRINTLBL.
400
500 100030 ENVIRONMENT DIVISION.
600
700 100040 INPUT-OUTPUT SECTION.
800
900 100050 FILE-CONTROL.
1000
1100 100060     SELECT PRINTER-FILE
1200
1300 100070         ASSIGN TO FORMATFILE-FONT.
1400
1500 100080 DATA DIVISION.
1600
1700 100090 FILE SECTION.
1800
1900 100100 FD PRINTER-FILE
2000
2100 100110    DATA RECORD IS REC1.
2200
2300 100120 01 REC1.
2400
2500 100130    COPY DDS-ALL-FORMATS OF FONT.
2600
2700 100140 PROCEDURE DIVISION.
2800
2900 100150 MAIN-PARA.
3000
3100 100160    OPEN OUTPUT PRINTER-FILE.
3200
3300 100170    WRITE REC1 FORMAT IS "REC1".
3400
3500 100180    CLOSE PRINTER-FILE.
3600
* * * * E N D   O F   S O U R C E * * * *
```

Notices

This information was developed for products and services offered in the U.S.A.
IBM may not offer the products, services, or features discussed in this document in
other countries. Consult your local IBM representative for information on the
products and services currently available in your area. Any reference to an IBM
product, program, or service is not intended to state or imply that only that IBM
product, program, or service may be used. Any functionally equivalent product,
program, or service that does not infringe any IBM intellectual property right may
be used instead. However, it is the user's responsibility to evaluate and verify the
operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter
described in this document. The furnishing of this document does not give you
any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY 10594
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM
Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

**The following paragraph does not apply to the United Kingdom or any other
country where such provisions are inconsistent with local law:**
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS
PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER
EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED
WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS
FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or
implied warranties in certain transactions, therefore, this statement may not apply
to you.

This information could include technical inaccuracies or typographical errors.
Changes are periodically made to the information herein; these changes will be
incorporated in new editions of the publication. IBM may make improvements
and/or changes in the product(s) and/or the program(s) described in this
publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for
convenience only and do not in any manner serve as an endorsement of those Web
sites. The materials at those Web sites are not part of the materials for this IBM
product and use of those Web sites is at your own risk.

Licensees of this program who wish to have information about it for the purpose
of enabling: (i) the exchange of information between independently created

programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator
3605 Highway 52 N
Rochester, MN 55901-7829
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming Interface Information

This publication is intended to help you to work with printing and spooling on OS/400. This publication documents General-Use Programming Interface and Associated Guidance Information provided by OS/400.

General-Use programming interfaces allow the customer to write programs that obtain the services of OS/400.

Trademarks

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

AD/Cycle
Advanced Function Presentation

Advanced Function Printing
AFCCU
AFP
AIX
APL2
Application System/400
AS/400
BookMaster
C/400
COBOL/400
e (logo)
Extended Services
GDDM
IBM
InfoColor
Infoprint
InfoWindow
Intelligent Printer Data Stream
IPDS
iSeries
Micro Channel
MVS
MVS/SP
OfficeVision/400
Operating System/400
OS/2
OS/390
OS/400
PrintManager
Print Services Facility
PS/2
Quietwriter
Redbooks
RPG/400
RS/6000
S/370
S/390
SAA
System/36
System/38
System/390
Systems Application Architecture

Java and all Java-based trademarks and logos are trademarks or registered
trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of
Microsoft Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks
of others.

Bibliography

The following is a list of manuals and topics in the iSeries Information Center that provide additional information about topics described or referred to in this manual. The manuals in this bibliography are listed with their full title and order number, but when referred to in text, a shortened version of the title is used.

- *About Type: Guide for Type Users (Current Release)* G544-3122.

  This publication introduces the use of typography in designing and printing documents with special emphasis using Advanced Function Printing fonts and printing systems.

- *About Type: Samples of 240-Pel Digitized Type* G544-3644.

  This manual provides type samples of 240-pel fonts.

- *About Type: Technical Reference for 240-Pel Digitized Type* , S544-3516.

  This manual provides information about the names, content and size of the character sets, code pages and coded fonts in the font licensed programs used in printing on 240-pel dots-per-inch (240-pel) non-impact printers supported by Advanced Function Printing (AFP) software.

- *About Type: Typographic Primer for Digitized Type* G544-3183.

  This publication introduces typographic concepts to show the reader how to create effective printed material with style and emphasis, and to illustrate how much value type characters from IBM's typographic fonts can add to business messages. This is primarily a marketing tool.

- Management Central.

  This topic in the iSeries Information Center gives system operators a fast path for system operator tasks on OS/400. It covers the basics of system startup and provides information about the system control panel, starting and stopping the system, using media, working with PTFs, and handling problems.

- Communication Configuration.

  Provided via PDF, using this manual, the user can configure the communications functions available with OS/400. It provides general configuration information, including detailed descriptions of network interface, line, controller, device, mode, and class-of-service descriptions, configuration lists, and connection lists.

- Networking.

  This topic in the iSeries Information Center contains information on configuring and using the Transmission Control Protocol/Internet Protocol (TCP/IP) and writing programs to the TCP/IP application program interface.

- DDS Reference: Printer Files.

  This topic in the iSeries Information Center provides detailed descriptions of the entries and keywords needed to describe database files (both logical and physical) and certain device files (for displays, printers, and intersystem communications function (ICF)) external to the user's programs.

- Database Management.

  This topic in the iSeries Information Center provides information about using files in application programs. This manual includes information on the following topics:

  - Fundamental structure and concepts of data management support on the system
  - Overrides and file redirection (temporarily making changes of files when an application program is run)
  - Copying files by using system commands to copy data from one place to another
  - Tailoring a system using double-byte data

- Local Device Configuration.

  Provided via PDF, this manual provides information on how to do an initial local hardware configuration and how to change that configuration. This manual also contains conceptual information about device configuration and planning information for device configuration on the 9406, 9404, and 9402 system units.

  The PDF includes additional information such as configuration terminology for lines, controllers, and devices. It also includes information on automatic configuration, how to do an initial configuration, how to change a configuration to suit the system, how to do local configuration including configuring ASCII

devices, and unique double-byte character set (DBCS) configuration information (as a separate appendix). Information about local, twinaxial, ASCII work station controllers, modems, and the devices that attach to these local workstation controllers is also included as well as forms for local work station attachment diagrams.

- *Graphic Object Content Architecture* SC31-6804.

  This manual explains the architecture of graphic objects for IPDS-capable devices.

- *IBM Enhanced 5250 Emulation Program User's Guide* G570-2221.

  This guide explains how to work with the IBM Enhanced 5250 Emulation Program.

- *IBM Print Services Facility for OS/2: Distributed Print Function Network Configuration Guide for OS/400* S544-3823.

  This guide explains how to configure the distributed print function and its associated hardware for use with Advanced Function Printing (AFP) on the iSeries server.

- *IBM Print Services Facility for OS/2: Getting Started* S544-3767.

  This manual explains how to plan for installation, install, and manage PSF for OS/2 in any host system environment that supports PSF for OS/2.

- *IBM Remote 5250 Emulation Program User's Guide* G570-2203.

  This guide explains how to work with the IBM Remote 5250 Emulation Program.

- *IBM Remote PrintManager User's Guide and Installation Guide* S544-3439.

  This manual explains how to install and configure the Remote PrintManager product for use with advanced function printing (AFP) on the iSeries server.

- *IBM's Guide for Using PMF With Type* S544-3648.

  This publication describes how to use fonts with PMF in MVS and VM. It was originally published as two separate books - one for each environment.

  This manual could help you in communicating with System/390 sites in transferring AFP resources to an iSeries server.

- CL Reference.

  This iSeries Information Center topic describes the OS/400 control language (CL) and its OS/400 commands. Each command description includes a syntax diagram; descriptions of parameters, default values, and keywords; and at least one example.

- GDDM Programming Guide.

  Provided via PDF, this guide provides information about using OS/400 graphical data display manager (GDDM) to write graphics application programs. GDDM allows you to add color and pictures to your application programs. This guide includes example programs and information to help you understand how the product fits into data processing systems.

- Work Management.

  The Work Management topic in the iSeries Information Center provides information about how to create and change a work management environment. Other topics include a description of tuning the system, collecting performance data including information on record formats and contents of the data being collected, working with system values to control or change the overall operation of the system, and a description of how to gather data to determine who is using the system and what resources are being used.

- System Operation

  This topic in the iSeries Information Center provides information about handling messages, working with jobs and printer output, device communications, working with support functions, cleaning up your system, and so on.

Index

Special Characters

Numerics

A

B

C

Readers' Comments — We'd Like to Hear from You

iSeries
Printer Device Programming
Version 5

Publication No. SC41-5713-05

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall satisfaction	☐	☐	☐	☐	☐

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	☐	☐	☐	☐	☐
Complete	☐	☐	☐	☐	☐
Easy to find	☐	☐	☐	☐	☐
Easy to understand	☐	☐	☐	☐	☐
Well organized	☐	☐	☐	☐	☐
Applicable to your tasks	☐	☐	☐	☐	☐

Please tell us how we can improve this book:

Thank you for your responses. May we contact you?   ☐ Yes   ☐ No

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name

Address

Company or Organization

Phone No.

IBM ®
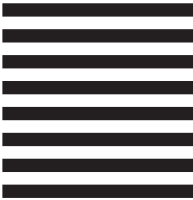
Printed in the United States of America
on recycled paper containing 10%
recovered post-consumer fiber.